

**APLIKASI SHALAT PADA *SMARTPHONE* BERBASIS  
ANDROID**

**TUGAS AKHIR**

Diajukan Sebagai Salah Satu Syarat  
Untuk Memperoleh Gelar Sarjana Teknik Pada  
Jurusan Teknik Informatika

Oleh :

**JOKO NURYANTO**  
**10751000257**



**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU  
PEKANBARU  
2013**

# **APLIKASI SHALAT PADA *SMARTPHONE* BERBASIS ANDROID**

**JOKO NURYANTO**

**10751000257**

Tanggal Sidang: 28 Oktober 2013

Periode Wisuda: Maret 2013

Jurusan Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Sultan Syarif Kasim Riau

## **ABSTRAK**

Untuk belajar shalat, sekarang ini kita dapat membaca buku, membuka internet, atau CD tutorial yang dimana penggunaanya harus berhubungan langsung dengan komputer. Dalam penelitian ini, penulis membahas tentang aplikasi shalat pada *smartphone* berbasis android, dimana penulis menggunakan metodologi *Rational Unified Process* (RUP). Aplikasi shalat pada *smartphone* sendiri sebenarnya sudah ada, tetapi belum bersifat multimedia. Pada penelitian ini, penulis membangun aplikasi shalat yang bersifat multimedia yang dimana nantinya *user* tidak hanya dapat membaca dan melihat gambar, tetapi *user* juga dapat mendengar suara bacaan doa dan melihat video shalat itu sendiri. Dari hasil pengujian yang telah penulis lakukan, maka dapat disimpulkan bahwa aplikasi ini dapat berjalan pada *smartphone* dengan OS android minimal versi 2.3 atau *gingerbread*, kemudian aplikasi ini juga *user friendly*.

**Kata kunci:** Android, Aplikasi Shalat, Shalat, *Smartphone*.

## KATA PENGANTAR

*Alhamdulillah Robbil'alamin*, penulis ucapkan puji syukur atas kehadiran Allah SWT, karena atas segala limpahan Rahmat dan Karunia-Nya yang diberikan sehingga penulis dapat menyelesaikan penulisan laporan tugas akhir ini. *Allahumma shalli'ala Muhammad wa'ala ali sayyidina Muhammad*, yang tidak lupa penulis haturkan juga untuk junjungan alam, kekasih Allah, Rasul Allah, dan tauladan kita sebagai umat muslim, yakni Nabi Muhammad SAW.

Laporan tugas akhir ini merupakan salah satu prasyarat untuk memenuhi persyaratan akademis dalam rangka meraih gelar kesarjanaan di Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Sultan Syarif Kasim Riau (UIN SUSKA Riau). Selama menyelesaikan tugas akhir ini, penulis telah banyak mendapatkan bantuan, bimbingan, dan petunjuk dari banyak pihak baik secara langsung maupun tidak langsung. Untuk itu dalam kesempatan ini penulis ingin mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Prof. Dr. H. M. Nazir, selaku Rektor Universitas Islam Negeri Sultan Syarif Kasim Riau.
2. Dra. Yenita Morena, M.Si, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau.
3. Ibu Elin Haerani, ST, M.Kom, selaku Ketua Jurusan Teknik Informatika.
4. Bapak Nazruddin Safaat H, MT, selaku Dosen Pembimbing Tugas Akhir. Terima kasih banyak nasehat, kebaikan-kebaikan, kesabaran, dan pengorbanan yang Bapak berikan kepada penulis.
5. Bapak Jasril, S.Si M.Sc, selaku dosen penguji 1, terimakasih atas ilmu-ilmunya, saran-sarannya, perbaikan-perbaikannya, dan masukan yang Bapak berikan untuk penyempurnaan laporan ini.
6. Bapak Febi Yanto, M.Kom, selaku dosen penguji 2, terimakasih juga untuk ilmu-ilmunya, saran-sarannya, perbaikan-perbaikannya, dan masukan yang Bapak berikan untuk penyempurnaan laporan ini juga.

7. Almarhum kedua Orang Tua penulis, yang telah mendahului kita semua, semoga semua amal kebbaikannya di terima Allah SWT, diberikan nikmat kubur, dan mendapatkan tempat di surga-NYA, amin.
8. Kedua Orang Tua Wali penulis yang telah penulis anggap sebagai Orang Tua sendiri, Bapak Adi Susanto dan Ibu Robi Yanti, milyaran terima kasih takkan cukup untuk membalas jasa-jasa beliau atas segala doa, dukungan, kesabaran, cinta kasih dan pengharapan yang beliau-beliau berikan, sehingga penulis mampu melaksanakan amanah mereka untuk menyelesaikan S1 dengan sebaik-baiknya.
9. Teman-teman TIF D angkatan 2007 yang masih seperjuangan, dan sahabat baik penulis, Jadno, Nuriyadi, Eryanto, Agustin Kurniasari, Zainal Mujahidin dan lain-lain yang selalu mendampingi, menyemangati, membantu penulis dalam menyelesaikan laporan, terimakasih atas segala hal yang kalian bagi kepada penulis.
10. Terimakasih kepada seseorang yang telah hadir disamping dan di hati penulis, yang tidak pernah berhenti memberi semangat dan selalu ada untuk penulis disaat senang maupun susah, semoga dimasa depan kita juga mampu menghadapi kehidupan bersama dalam susah maupun senang sampai ajal menjemput kita, dan memisahkan kebersamaan kita didunia untuk kembali bersama di surga, Amin.
11. Dan terakhir, terimakasih penulis ucapkan untuk Teknik Informatika, Fakultas Sains dan Teknologi, UIN Sultan Syarif Kasim Riau serta pihak-pihak lain yang tidak dapat penulis sebutkan satu persatu. Terimakasih banyak atas bantuan dan dukungannya yang berharga.

Akhirnya, penulis menyadari dalam penulisan laporan ini masih terdapat banyak kekurangan. Oleh karena itu, saran dan kritik sangat penulis harapkan untuk kemajuan penulis secara pribadi dan bersama. Terimakasih.

Pekanbaru, 28 Oktober 2013

Penulis

# DAFTAR ISI

## Halaman

|   |      |
|---|------|
| HALAMAN JUDUL LAPORAN.....                | i    |
| LEMBAR PERSETUJUAN.....                   | ii   |
| LEMBAR PENGESAHAN .....                   | iii  |
| LEMBAR HAK ATAS KEKAYAAN INTELEKTUAL..... | iv   |
| LEMBAR PERNYATAAN.....                    | v    |
| LEMBAR PERSEMBAHAN .....                  | vi   |
| ABSTRAK.....                              | vii  |
| <i>ABSTRACT</i> .....                     | viii |
| KATA PENGANTAR .....                      | ix   |
| DAFTAR ISI.....                           | xi   |
| DAFTAR GAMBAR .....                       | xiv  |
| DAFTAR TABEL.....                         | xvi  |
| DAFTAR LAMPIRAN.....                      | xvii |
| BAB I PENDAHULUAN.....                    | I-1  |
| 1.1 Latar Belakang .....                  | I-1  |
| 1.2 Rumusan Masalah .....                 | I-2  |
| 1.3 Batasan Masalah .....                 | I-2  |
| 1.4 Tujuan .....                          | I-3  |
| 1.5 Sistematika Penulisan .....           | I-3  |
| BAB II LANDASAN TEORI .....               | II-1 |
| 2.1 Shalat .....                          | II-1 |
| 2.1.1 Dalil yang Mewajibkan Shalat .....  | II-1 |
| 2.1.2 Tata Cara Sholat .....              | II-2 |
| 2.1.3 Bacaan Sholat .....                 | II-2 |
| 2.2 Teknologi <i>Mobile</i> .....         | II-3 |
| 2.3 Android .....                         | II-4 |
| 2.3.1 Sejarah Android .....               | II-4 |

|  |       |
|--|-------|
| 2.3.2 Arsitektur Android .....                                       | II-5  |
| 2.3.3 Komponen Aplikasi .....  | II-6  |
| 2.3.4 Kelebihan <i>Platform</i> Android .....                        | II-7  |
| 2.4 <i>Unified Modelling Language</i> (UML).....                     | II-7  |
| 2.4.1 <i>Use Case</i> Diagram .....                                  | II-8  |
| 2.4.2 <i>Class</i> Diagram .....                                     | II-9  |
| 2.4.3 <i>Activity</i> Diagram .....                                  | II-9  |
| 2.4.4 <i>Sequence</i> Diagram.....                                   | II-9  |
| 2.5 <i>Software</i> dan <i>Tool</i> .....                            | II-10 |
| 2.5.1 <i>Java</i> .....  | II-10 |
| 2.5.2 <i>Eclipse</i> .....   | II-10 |
| 2.5.3 <i>ADT</i> .....   | II-11 |
| 2.5.4 <i>Android SDK</i> .....                                       | II-12 |
| 2.5.5 <i>Adobe Photoshop</i> .....                                   | II-12 |
| 2.5.6 <i>3D Studio Max</i> .....                                     | II-14 |
| 2.5.7 <i>Adobe Soundbooth</i> .....                                  | II-15 |
| 2.6 <i>Rational Unified Process</i> (RUP) .....                      | II-16 |
| 2.6.1 Fase RUP .....   | II-17 |
| 2.6.1.1 Fase <i>Inception</i> .....                                  | II-17 |
| 2.6.1.2 Fase <i>Elaboration</i> .....                                | II-17 |
| 2.6.1.3 Fase <i>Construction</i> .....                               | II-18 |
| 2.6.1.4 Fase <i>Transition</i> .....                                 | II-18 |
| BAB III METODOLOGI PENELITIAN.....                                   | III-1 |
| 3.1 Tahapan Penelitian .....   | III-1 |
| 3.2 Tahapan pengembangan <i>Rational Unified Process</i> (RUP) ..... | III-2 |
| 3.2.1 Fase <i>Inception</i> .....                                    | III-2 |
| 3.2.2 Fase <i>Elaboration</i> .....                                  | III-2 |
| 3.2.3 Fase <i>Construction</i> .....                                 | III-3 |
| 3.2.3 Fase <i>Transition</i> .....                                   | III-3 |
| BAB IV ANALISIS DAN PERANCANGAN .....                                | IV-1  |
| 4.1 Fase <i>Inception</i> dan <i>Elaboration</i> .....               | IV-1  |

|  |       |
|--|-------|
| 4.1.1 Gambaran Umum Sistem .....                                   | IV-1  |
| 4.1.2 Deskripsi Kebutuhan Aplikasi .....                           | IV-1  |
| 4.1.2.1 Aplikasi yang akan dibangun .....                          | IV-2  |
| 4.1.2.2 Analisa fungsional .....                                   | IV-3  |
| 4.1.3 Perancangan Sistem .....                                     | IV-3  |
| 4.1.3.1 Pemodelan UML ( <i>Unified Modeling Language</i> ) ..      | IV-3  |
| 4.1.3.1.1 <i>Usecase</i> Diagram .....                             | IV-4  |
| 4.1.3.1.2 <i>Class</i> Diagram .....                               | IV-6  |
| 4.1.3.1.3 <i>Activity</i> Diagram .....                            | IV-7  |
| 4.1.3.1.4 <i>Sequence</i> Diagram .....                            | IV-11 |
| 4.2 Fase <i>Elaboration</i> .....                                  | IV-12 |
| 4.2.1 Perancangan Struktur Menu Sistem .....                       | IV-12 |
| 4.2.2 Perancangan <i>Interface</i> .....                           | IV-13 |
| 4.2.3 Perancangan Animasi 3D .....                                 | IV-16 |
| BAB V IMPLEMENTASI DAN PENGUJIAN .....                             | V-1   |
| 5.1 Fase <i>Construction</i> .....                                 | V-1   |
| 5.1.1 Implementasi .....   | V-1   |
| 5.1.1.1 Lingkungan Implementasi .....                              | V-1   |
| 5.1.2 Implementasi Kelas .....                                     | V-2   |
| 5.1.3 Hasil Implementasi Aplikasi pada <i>Device</i> Android ..... | V-7   |
| 5.2 Fase <i>Transition</i> .....                                   | V-10  |
| 5.2.1 Pengujian Aplikasi Shalat Android .....                      | V-10  |
| 5.2.2 Pengujian Kepada Pengguna .....                              | V-16  |
| 5.2.2.1 Hasil Kuisisioner Responden Siswa .....                    | V-16  |
| 5.2.2.2 Hasil Kuisisioner Responden Muallaf .....                  | V-16  |
| BAB VI PENUTUP .....   | VI-1  |
| 6.1 Kesimpulan .....   | VI-1  |
| 6.2 Saran .....  | VI-1  |
| DAFTAR PUSTAKA   |       |
| LAMPIRAN   |       |
| RIWAYAT HIDUP  |       |

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Shalat adalah sebuah kewajiban utama bagi setiap muslim untuk menentukannya. Shalat adalah tiang dari agama dan amal perbuatan seorang hamba yang pertama kali dihisab pada hari akhir nanti tidak lain adalah shalat. Kewajiban shalat ini juga telah difirmankan Allah SWT dalam salah satu surat Al-Qur'an yaitu dalam surat Al-Ankabut ayat 45 yang artinya "*Kerjakanlah shalat, sesungguhnya shalat itu mencegah perbuatan yang jahat dan mungkar*". Setiap umat muslim diwajibkan untuk mengikuti tata cara shalat Nabi Muhammad SAW, seperti yang telah beliau sabdakan kepada kita semua. Beliau bersabda : "*Shalatlah kamu sekalian, sebagaimana melihatku shalat*". (Tata cara shalat sesuai dengan tuntunan Rasulullah SAW, Syaikh Muhammad Nashiruddin al-Albani, 2012)

Oleh karena itu kita harus berpegang pada hadits Rasulullah SAW yang mengajarkan tentang tata cara shalat sesuai dengan tuntunannya. Sudah menjadi kewajiban setiap muslim untuk berbagi ilmu, terutama dalam tata cara shalat yang baik dan benar sesuai petunjuk Rasulullah SAW. Saat ini tata cara shalat sudah dibuat dalam berbagai media dengan semua teknologi dan peralatan yang mendukungnya yaitu mulai dari buku, artikel, CD tutorial, aplikasi shalat berbasis *desktop* atau *mobile*, serta dari internet yang mengharuskan pengguna berhubungan langsung dengan komputer maupun perangkat yang lain.

Aplikasi shalat pada perangkat *mobile* sendiri juga telah ada, tetapi masih memiliki kekurangan yaitu tidak bersifat multimedia. Aplikasi shalat pada perangkat *mobile* berbasis android yang telah penulis lihat yaitu aplikasi "*Learn Shalat*". Dalam aplikasi ini, fitur yang diberikan masih berupa narasi dan gambar



statis. Sementara beberapa gerakan atau posisi akan sulit untuk dipahami apabila hanya disajikan dalam bentuk gambar dan narasi saja. Dari sini pentingnya penulis untuk menambah beberapa fitur multimedia. Seiring dengan perkembangan teknologi komunikasi, dan banyaknya orang yang telah menggunakan *smartphone* sebagai perangkat utama dalam kehidupan sehari-hari, serta banyaknya vendor yang menggunakan android sebagai sistem operasi pada produk *smartphone* mereka ([www.idc.com](http://www.idc.com)), maka berdasarkan hal tersebut penulis mengambil judul “*Aplikasi Shalat pada Smartphone berbasis Android*”.

Aplikasi ini merupakan pengembangan dari aplikasi yang sudah ada, sehingga nantinya akan bersifat multimedia karena penulis akan membangun sebuah aplikasi dengan fitur-fitur yang bersifat multimedia, dengan adanya aplikasi ini pengguna tidak hanya dapat melihat gambar tetapi juga dapat melihat gerakan dan mendengar suara dari aplikasi tersebut. Pesatnya perkembangan teknologi *smartphone* dan juga sistem operasi yang digunakan sekarang ini, untuk membuat atau membangun sebuah aplikasi yang bersifat multimedia tidaklah sulit, karena fitur-fitur yang disediakan oleh *smartphone* itu sendiri sudah sangat mendukung.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang diatas, maka penulis dapat merumuskan “Bagaimana membuat aplikasi shalat pada *smartphone* berbasis android?”.

## **1.3 Batasan Masalah**

Dalam pembuatan aplikasi ini, penelitian dilakukan dengan batasan-batasan sebagai berikut:

1. Tata cara shalat serta penjelasannya mereferensi dari buku dengan judul “ Sifat Shalat Nabi, Tata Cara Shalat Sesuai Dengan Tuntunan Rasulullah SAW “ karangan Syaikh Muhammad Nashiruddin al-Albani tahun 2012.

2. Dalam pembuatan aplikasi shalat ini, penulis mengacu pada mazhab *Ahlus Sunnah Wal Jama'ah* sesuai dengan referensi buku yang digunakan oleh penulis.

#### **1.4 Tujuan Penelitian**

Tujuan yang ingin dicapai dari tugas akhir ini ialah untuk membuat sebuah aplikasi shalat pada *smartphone* berbasis android yang nantinya dapat bermanfaat bagi banyak orang.

#### **1.5 Sistematika Penulisan**

### **BAB I Pendahuluan**

Pada bab ini berisi penjelasan Latar Belakang, Rumusan Masalah, Batasan Penelitian, Tujuan Penelitian, dan Sistematika Penulisan.

### **BAB II Landasan Teori**

Membahas teori-teori pendukung penulisan tugas akhir ini. Seperti tentang pengertian shalat, android dan lain sebagainya yang berhubungan dengan penulisan tugas akhir ini.

### **BAB III Metodologi Penelitian**

Bab ini membahas langkah-langkah yang dilaksanakan dalam proses penelitian, yaitu menggunakan metodologi *Rational Unified Process* (RUP). RUP merupakan salah satu metodologi pengembangan perangkat lunak berorientasi objek dan implementasi beserta pengujian.

### **BAB IV Analisa dan Perancangan**

Bab ini berisi pembahasan mengenai kebutuhan sistem, yang terdiri dari : UML, *User interface*, perancangan menggunakan pendekatan berorientasi objek sesuai dengan metodologi RUP pada bab sebelumnya.

## **BAB V Implementasi dan Pengujian**

Bab ini menjelaskan mengenai Implementasi penelitian yang dilakukan dengan melakukan berbagai evaluasi dan perbaikan yang dirasa perlu berdasarkan hasil penelitian.

## **BAB VI Penutup**

Bab ini membahas mengenai kesimpulan yang dapat diambil dari penelitian dan saran yang diperoleh untuk pengembangan aplikasi lebih lanjut.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Shalat**

Shalat menurut makna adalah doa. Yang dimaksud disini ialah ibadah yang tersusun dari beberapa perkataan dan perbuatan. Dimulai dengan takbir dan diakhiri dengan salam, serta memenuhi syarat yang telah ditentukan. ( Tuntunan Pelajaran Shalat Lengkap, H. Fachrurozi).

##### **2.1.1 Dalil yang Mewajibkan Shalat**

Dalil yang mewajibkan shalat banyak sekali, baik dalam Al-Qur'an maupun Hadits Nabi Muhammad SAW. Berikut contoh dalil dalam Al-Qur'an yang mewajibkan shalat antara lain :

Q.S Al-Baqarah : 43 yang artinya :

*“Dan dirikanlah shalat dan keluarkanlah zakat dan tunduklah bersama-sama orang-orang yang rukuk”*

Q.S Al-Ankabut : 45 yang artinya :

*“Kerjakanlah shalat, sesungguhnya shalat itu mencegah perbuatan yang jahat dan mungkar”*

Dan hendaklah perintah shalat ini ditanamkan ke dalam hati dan jiwa anak-anak dengan cara pendidikan yang cermat dan dilakukan sejak kecil. Berikut contoh dalil dalam hadits Nabi Muhammad SAW yang artinya :

*“Perintahkanlah anak-anakmu mengerjakan shalat di waktu usia mereka meningkat tujuh tahun, dan pukullah kalau enggan melakukan shalat di waktu mereka meningkat usia sepuluh tahun.”* ( HR. Abu Dawud)

Selain dalil diatas, masih banyak lagi dalil yang mewajibkan umat islam untuk mendirikan shalat. Dan dalam islam, shalat menempati posisi penting dan strategis karena merupakan salah satu rukun islam yang menjadi pembatas apakah

orang itu mukmin atau kafir. (Pendidikan Agama Islam, Ika Setiyani, Dica Lanita, Ismunajab, 2010)

### 2.1.2 Tata Cara Sholat

#### 1. Berdiri Tegak

Rasullullah SAW berdiri tegak ketika mengerjakan shalat fardhu maupun sunnah. Beliau melakukannya untuk melaksanakan perintah Allah SWT, yang artinya : *“Berdirilah untuk Allah (dalam shalatmu) dengan khusyu’.”* (QS.Al-Baqarah:238). Namun apabila dalam keadaan sakit kita dibolehkan mengerjakan shalat sambil duduk dan jika tidak sanggup duduk maka dibolehkan dengan berbaring. Hal ini sesuai dengan hadits Rasulullah SAW yang artinya: *“Shalatlah sambil berdiri, apabila kamu tidak sanggup, maka sambil duduk, apabila kamu tidak sanggup juga maka sambil berbaring dengan sisi tubuh”.* (HR.Al-Bukhari, Abu Dawud dan Ahmad). Untuk tata cara shalat yang selanjutnya dapat dilihat pada lampiran A.

### 2.1.3 Bacaan Sholat

#### 1. Takbir

Dalam mengerjakan shalat, Rasullullah SAW selalu memulai shalatnya dengan ucapan :

**‘Allahu Akbar’.** (HR. Muslim dan Ibnu Majah).

Hal ini seperti yang telah diperintahkan beliau dalam sabdanya yang artinya: *“Sesungguhnya shalat seseorang tidaklah sempurna hingga ia berwudhu’, dan melakukan wudhu’ sesuai ketentuannya, kemudian ia mengucapkan ‘Allahu Akbar’.”* (HR.At- Thabrani).

Setelah takbir, beliau memulai dengan membaca doa iftitah. Doa iftitah yang dibaca beliau bervariasi. Didalam doa tersebut beliau memuji Allah SWT , mengagungkan dan menyanjungNya. salah satu doa iftitah yang beliau baca dalam shalat fardhu yaitu:

**‘Subkhaanakallaahumma wabikhamdika watabaarakasmuka wataka’aala jadduka walaa ilaaha ghairuka’.** (HR. Abu Dawud dan Al-Hakim)

Kemudian beliau membaca surat Al-Fatihah dengan memutus bacaan ayat demi ayat hingga akhir surat. Dan seperti itulah bacaan Al-Qur'an Rasullullah SAW seluruhnya. Beliau berhenti disetiap akhir ayat dan tidak menyambung dengan ayat setelahnya. (HR. Abu Dawud, As-Sahmi, Al-Hakim dan Adz-Dzahbi). Untuk bacaan shalat yang selanjutnya dapat dilihat pada lampiran A.

## **2.2 Teknologi *Mobile***

Teknologi perangkat *smartphone* semakin berkembang pesat diseluruh dunia. Perkembangan yang terjadi ini akibat kebutuhan manusia akan perangkat *smartphone* tersebut sudah sangat menyatu dengan kehidupan sehari-hari. Perkembangan yang paling pesat yang terjadi pada teknologi *mobile* ada pada sistem operasi yang digunakan pada *smartphone* tersebut, dan karena itulah maka banyak berkembang aplikasi *mobile learning*.

*Mobile Learning (m-Learning)* didefinisikan oleh Clark Quin (2006) sebagai :

*“The intersection of mobile computing and e-learning: accessible resources wherever you are, strong search capabilities, rich interaction, powerful support for effective learning, and performance-based assessment. Elearning independent of location in time or space”.*

Merujuk dari definisi tersebut maka, *mobile learning* adalah model-model pembelajaran yang memanfaatkan teknologi informasi dan komunikasi. Pada konsep pembelajaran tersebut *m-learning* membawa manfaat ketersediaan materi ajar yang dapat diakses setiap saat dan visualisasi materi yang menarik. Dan hal penting yang perlu diperhatikan bahwa tidak setiap materi pengajaran cocok memanfaatkan m-learning.

*Mobile Learning* dapat didefinisikan sebagai suatu fasilitas atau layanan yang memberikan informasi elektronik secara umum kepada pembelajar dan *content* yang edukasional yang membantu pencapaian pengetahuan tanpa mempermasalahkan lokasi dan waktu. Sistem *m-learning* ini memanfaatkan mobilitas dari perangkat mobile, seperti *handpone* dan PDA, untuk memberikan

suatu fungsi pembelajaran yang dapat dilakukan dimana pun dan kapan pun. (M.Irfan Aripurnama, 2011)

Pemrograman yang semakin berkembang dan harga yang semakin terjangkau dari *smartphone*, menjadikan aplikasi *mobile* ini menjadi salah satu pilihan untuk media pembelajaran.

## **2.3 Android**

*Android* adalah sistem operasi untuk telepon seluler yang berbasis Linux. *Android* menyediakan *platform* terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. *Android* memiliki berbagai keunggulan sebagai *software* yang memakai basis kode komputer yang bisa didistribusikan secara terbuka (*open source*) sehingga pengguna bisa membuat aplikasi baru di dalamnya.

*Android* merupakan suatu *software* (perangkat lunak) yang digunakan pada *mobile device* (perangkat berjalan) yang meliputi Sistem Operasi, Middleware dan Aplikasi Inti. *Android* SDK menyediakan alat dan API yang diperlukan untuk memulai pengembangan aplikasi pada platform *Android* menggunakan bahasa pemrograman Java, yaitu kode Java yang terkompilasi dengan data dan *file resources* yang dibutuhkan aplikasi dan digabungkan oleh *aapt tools* menjadi paket Android. File tersebut ditandai dengan ekstensi *.apk*. File inilah yang didistribusikan sebagai aplikasi dan diinstall pada perangkat *mobile*.

### **2.3.1 Sejarah Android**

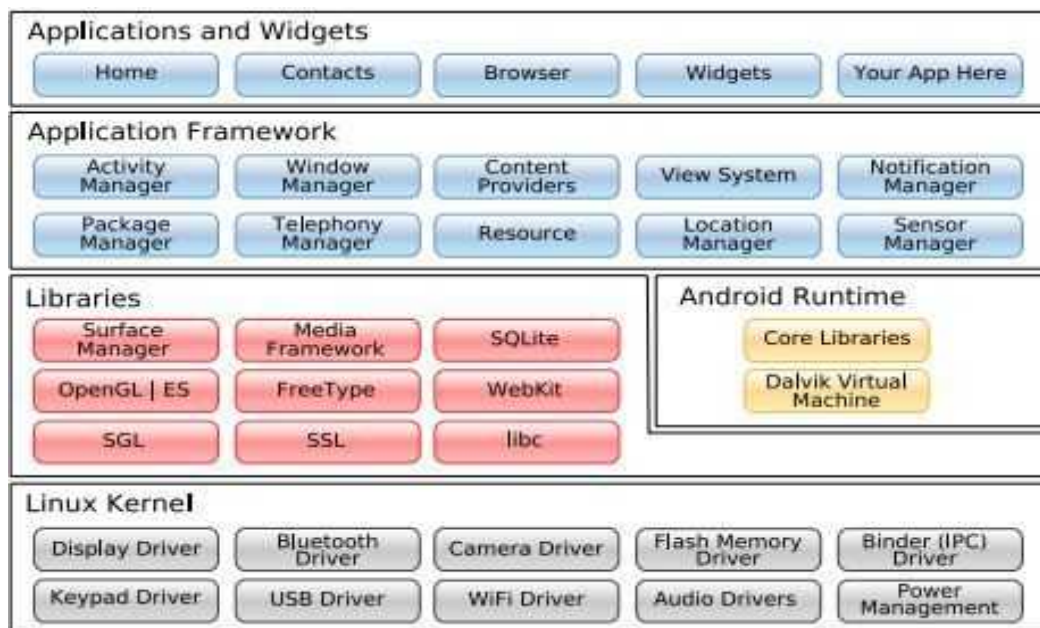
Pada tahun 2005 Google mengakuisisi *Android* Inc yang pada saat itu dimotori oleh Andy Rubin, Rich Miner, Nick Sears dan Chris White. Kemudian pada tahun itu juga memulai membangun platform *Android* secara intensif. Kemudian pada tanggal 5 November 2007 *Android* pertama kali diluncurkan, dan *smartphone* pertama yang menggunakan sistem operasi *Android* dikeluarkan oleh T-Mobile dengan sebutan G1 pada bulan September 2008.

Hingga saat ini *Android* telah merilis beberapa versi *Android* untuk menyempurnakan versi sebelumnya. Selain berdasarkan penomoran, pada setiap versi *Android* terdapat kode nama berdasarkan nama-nama kue. Hingga saat ini

sudah terdapat beberapa versi yang telah diluncurkan, diantaranya: versi 1.5 dirilis pada 30 April 2009 diberi nama *Cupcake*, versi 1.6 dirilis pada 15 September 2009 diberi nama *Donut*, versi 2.0/2.1 dirilis pada 26 Oktober 2009 diberi nama *Éclair*, versi 2.2 dirilis pada bulan Mei 2010 diberi nama *Froyo* dan versi 2.3 dirilis pada Desember 2010 yang diberi nama *Gingerbread*, versi 3.0 dirilis pada Februari 2011 yang diberi nama *Honeycomb*, dan versi 4.0 dirilis pada November 2011 dengan nama *Android Ice Cream sandwich*.

### 2.3.2 Arsitektur Android

*Android* merupakan platform yang *open source* sehingga banyak pihak dapat mengembangkan aplikasi dengan platform android. Gambar di bawah ini merupakan diagram arsitektur platform android.



**Gambar 2.1 Arsitektur Android**

1. *Application and Wigdetd* (paling atas), user hanya berinteraksi pada aplikasi seperti *download* dan *install*.
2. *Application Framework* (ke dua dari atas) adalah layer bagi para pembuat aplikasi.
3. *Libraries* dan *android runtime* adalah layer bagi aplikasi yang ada database seperti *sql-lite*.
4. *Linux Kernel* merupakan layer untuk *root*.



### 2.3.3 Komponen Aplikasi

Aplikasi *Android* ditulis dalam bahasa pemrograman Java. Kode Java dikompilasi bersama dengan data file *resource* yang dibutuhkan oleh aplikasi, prosesnya di paket oleh *tools* yang dinamakan *atp tools* kedalam paket *Android*, sehingga menghasilkan file dengan ekstensi *apk*. Ada 4 jenis komponen pada aplikasi *android*, yaitu:

1. *Activities*

Suatu *Activity* akan menyajikan *user interface* (UI) kepada pengguna, sehingga pengguna dapat melakukan interaksi. Satu *activity* biasanya akan dipakai untuk menampilkan aplikasi atau yang bertindak sebagai *user interface* (UI) saat aplikasi diperlihatkan kepada *user*. Untuk pindah dari satu *activity* ke *activity* lain, kita dapat melakukan dengan satu even, misalnya klik tombol, memilih opsi atau menggunakan *triggers* tertentu.

2. *Service*

*Service* tidak memiliki *Graphic User Interface* (GUI), tetapi *service* berjalan secara background. Komponen *service* diproses tidak terlihat, memperbaharui sumber data dan menampilkan notifikasi. *Service* digunakan untuk melakukan pengolahan data yang terus diproses, bahkan ketika *activity* tidak aktif

3. *Broadcast Receiver*

*Broadcast receiver* berfungsi menerima dan bereaksi untuk menyampaikan notifikasi. *Broadcast receiver* tidak memiliki *user interface*, tetapi memiliki sebuah *activity* untuk merespon informasi yang diterima atau menggunakan *notification manager* untuk memberi tahu pengguna, seperti lampu latar atau getaran.

4. *Content Provider*

*Content provider* membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain. Data disimpan dalam file seperti database SQLite. *Content provider* menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu *activity*, misalnya ketika kita menggunakan aplikasi

yang membutuhkan peta atau aplikasi yang membutuhkan untuk mengakses data kontak dan navigasi, maka disinilah fungsi dari content provider.

#### **2.3.4 Kelebihan Platform Android**

Persaingan *platform* atau sistem operasi semakin ketat, ini dapat dilihat dari banyaknya sistem operasi yang ada seperti, *Symbian*, *BlackBerry*, *Java Mobile Edition* dan *Linux Mobile (LiM0)*. Namun, *Android* adalah yang pertama menggabungkan hal seperti berikut :

1. Keterbukaan, Bebas pengembangan tanpa dikenakan biaya terhadap sistem karena berbasis Linux dan *open source*. Pembuat perangkat menyukai hal ini karena dapat membangun *platform* yang sesuai yang diinginkan tanpa harus membayar *royalty*.
2. Arsitektur komponen dasar *Android* terinspirasi dari teknologi internet *Mashup*. Bagian dalam sebuah aplikasi dapat digunakan oleh aplikasi lainnya, bahkan dapat diganti dengan komponen lain yang sesuai dengan aplikasi yang dikembangkan.
3. Banyak dukungan *service*, kemudahan dalam menggunakan berbagai macam layanan pada aplikasi seperti penggunaan layanan pencarian lokasi, database SQL, browser dan penggunaan peta. Semua itu sudah tertanam pada *Android* sehingga memudahkan dalam pengembangan aplikasi.
4. Siklus hidup aplikasi diatur secara otomatis, setiap program terjaga antara satu sama lain oleh berbagai lapisan keamanan, sehingga kerja sistem menjadi lebih stabil.
5. Dukungan grafis dan suara terbaik, dengan adanya dukungan 2D grafis dan animasi yang diilhami oleh *Flash* menyatu dalam 3D menggunakan *OpenGL* memungkinkan membuat aplikasi maupun game yang berbeda.
6. Portabilitas aplikasi, aplikasi dapat digunakan pada perangkat yang ada saat ini maupun yang akan datang.

#### **2.4 Unified Modelling Language (UML)**

UML (*Unified Modeling Language*) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan,

membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (*Object-Oriented*). *UML* sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software*. (<http://www.omg.org>)

Dalam pemodelan *UML*, ada delapan diagram yang digunakan dalam pemodelan diantaranya adalah.

- 1) *use case diagram*
- 2) *class diagram*
- 3) *statechart diagram*
- 4) *activity diagram*
- 5) *sequence diagram*
- 6) *collaboration diagram*
- 7) *component diagram*
- 8) *deployment diagram*

Berikut ini adalah beberapa penjelasan dari diagram pada *UML* yang akan penulis gunakan dalam membangun aplikasi nantinya.

#### **2.4.1 Use Case Diagram**

*Use case diagram* merupakan gambaran fungsionalitas yang pada sebuah sistem. *Use case* menggambarkan apa saja yang dapat dilakukan sistem bukan bagaimana sistem tersebut bekerja. *Use case* menggambarkan interaksi antara sistem dan aktor pada saat sistem digunakan. *Use case* merupakan sebuah pekerjaan yang dapat dilakukan oleh aktor terhadap sistem, misalnya *login* ke sistem, atau membuat daftar belanja dan sebagainya. Aktor pada *use case* adalah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan tertentu (Sri Dharwiyanti dan Romi, 2003).

### 2.4.2 Class Diagram

*Class Diagram* menggambarkan sebuah spesifikasi yang menjelaskan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan atau atribut suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut atau yang biasa disebut fungsi (Sri Dharwiyanti dan Romi, 2003).

*Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain.

### 2.4.3 Activity Diagram

*Activity diagram* menggambarkan beberapa aliran aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

*Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan *behaviour* internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum (Sri Dharwiyanti dan Romi, 2003).

### 2.4.4 Sequence Diagram

*Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem yang berupa *message* yang digambarkan prosesnya mulai dari pengguna, *display*, hingga proses dan data terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

*Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas

tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan (Sri Dharwiyanti dan Romi, 2003).

## **2.5 Software dan Tool**

Untuk membangun sebuah aplikasi android, maka penulis nantinya menggunakan beberapa *software* dan *tool* yang akan digunakan yaitu Java, Eclipse dan ADT, Android SDK, 3D Max dan Adobe Soundbooth.

### **2.5.1 Java**

Aplikasi android merupakan aplikasi yang dikembangkan dengan berbasis java. Java merupakan bahasa pemrograman yang dapat dijalankan di berbagai komputer dan telepon genggam. Aplikasi-aplikasi berbasis java umumnya dikompilasi ke dalam p-code (bytecode) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM). Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (general purpose), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi java mampu berjalan di beberapa platform sistem operasi yang berbeda, java dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Saat ini java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web.

### **2.5.2 Eclipse**

*Eclipse* adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua *platform* (*platform independent*) yang digunakan sebagai *compiler* dalam membuat aplikasi android. *Eclipse* awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak IBM *Visual Age for Java 4.0*. Pada saat ini *eclipse* merupakan salah satu IDE yang paling banyak digunakan dalam mengembangkan perangkat lunak. Hal ini dikarenakan *eclipse* merupakan perangkat yang *open source* dan gratis. Kelebihan *eclipse* diantaranya yaitu:

1. Multi-platform yaitu target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Juga Mac OS X.
2. Multi-language yaitu eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
3. Multi-role yaitu selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

### 2.5.3 ADT

*Android Development Tool* (ADT) adalah *plugin* yang didesain untuk IDE Eclipse yang memberikan kita kemudahan dalam mengembangkan aplikasi android dengan menggunakan IDE Eclipse. Dengan menggunakan ADT untuk Eclipse, maka akan memudahkan kita dalam membuat aplikasi project android, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainnya, begitu juga kita dapat melakukan *running* aplikasi menggunakan Android SDK melalui Eclipse. Dengan ADT juga, kita dapat melakukan pembuatan *package* android (.apk) yang digunakan untuk distribusi aplikasi android yang kita rancang dan bangun.

APK adalah paket aplikasi android (*android package*). APK umumnya digunakan untuk menyimpan sebuah aplikasi atau program yang akan dijalankan pada perangkat Android. APK pada dasarnya seperti zip file, karena berisi dari kumpulan file. APK dapat diperoleh melalui berbagai metode, seperti menginstal sebuah aplikasi melalui *market*, download dari sebuah situs web, atau membuat sendiri dengan bahasa java. Jika Anda memiliki file APK pada komputer anda, dan ingin menginstalnya pada telepon, maka anda dapat menjalankan perintah '*adb install apkgname.apk*' untuk install aplikasi melalui USB ke *smartphone* Anda.

Mengembangkan aplikasi android dengan menggunakan ADT di eclipse sangat dianjurkan dan sangat mudah untuk memulai mengembangkan aplikasi android yang kita inginkan. Semakin tinggi platform android yang kita gunakan, dianjurkan menggunakan ADT yang terbaru.

#### **2.5.4 *Android SDK***

Android SDK merupakan *tool API (Application Programming Interface)* yang diperlukan untuk mulai mengembangkan aplikasi pada *platform android* menggunakan bahasa pemrograman java. Android merupakan *subset* perangkat lunak untuk ponsel yang meliputi system operasi, *middleware* dan aplikasi kunci yang di *release* oleh google. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform android* menggunakan bahasa pemrograman java.

#### **2.5.5 *Adobe Photoshop***

*Adobe Photoshop*, atau biasa disebut *Photoshop*, adalah perangkat lunak editor citra buatan *Adobe Systems* yang dikhususkan untuk pengeditan foto/gambar dan pembuatan efek. Perangkat lunak ini banyak digunakan oleh fotografer digital dan perusahaan iklan sehingga dianggap sebagai pemimpin pasar (*market leader*) untuk perangkat lunak pengolah gambar/foto, dan, bersama *Adobe Acrobat*, dianggap sebagai produk terbaik yang pernah diproduksi oleh *Adobe Systems*. Versi kedelapan aplikasi ini disebut dengan nama *Photoshop CS (Creative Suite)*, versi sembilan disebut *Adobe Photoshop CS2*, versi sepuluh disebut *Adobe Photoshop CS3*, versi kesebelas adalah *Adobe Photoshop CS4*, versi keduabelas adalah *Adobe Photoshop CS5*, dan versi yang terakhir (ketigabelas) adalah *Adobe Photoshop CS6*.

*Photoshop* tersedia untuk *Microsoft Windows*, *Mac OS X*, dan *Mac OS*; versi 9 ke atas juga dapat digunakan oleh sistem operasi lain seperti *Linux* dengan bantuan perangkat lunak tertentu seperti *Cross Over*. Meskipun pada awalnya *Photoshop* dirancang untuk menyunting gambar untuk cetakan berbasis-kertas. *Photoshop* yang ada saat ini juga dapat digunakan untuk memproduksi gambar

untuk *World Wide Web*. Beberapa versi terakhir juga menyertakan aplikasi tambahan *Adobe ImageReady*, untuk keperluan tersebut.

Photoshop juga memiliki hubungan erat dengan beberapa perangkat lunak penyunting media, animasi, dan authoring buatan-Adobe lainnya. File format asli *Photoshop*, *.PSD*, dapat diekspor ke dan dari *Adobe ImageReady*. *Adobe Illustrator*, *Adobe Premiere Pro*, *After Effects* dan *Adobe Encore DVD* untuk membuat DVD profesional, menyediakan penyuntingan gambar non-linear dan layanan special effect seperti background, tekstur, dan lain-lain untuk keperluan televisi, film, dan situs web. Sebagai contoh, *Photoshop CS* dapat digunakan untuk membuat menu dan tombol (*button*) DVD. *Photoshop* dapat menerima penggunaan beberapa model warna:

- *RGB color model*
- *Lab color model*
- *CMYK color model*
- *Grayscale*
- *Bitmap*
- *Duotone*

Versi terbarunya, yang dirilis pada tahun 2005, adalah versi 9. Program ini dipasarkan dengan nama "*Photoshop CS2*". "*CS*" merefleksikan integrasi produk *Photoshop* dengan aplikasi "*Creative Suite*" buatan Adobe dan disebut "*2*" karena program ini adalah versi rilis ke-2 sejak Adobe mengintegrasikan kedua produknya. Ada beberapa tambahan pada *Photoshop CS2* seperti *multiple layer selecting* dan "*warp*," versi kurva dari *transform tool* dan *color replacement tool*, yang sebelumnya hadir sebagai *plug-in* 8BF.

*Photoshop* memiliki kemampuan untuk membaca dan menulis gambar berformat raster seperti *.png*, *.gif*, *.jpeg*, dan lain-lain. *Photoshop* juga memiliki beberapa format file khas:

- *PSD (Photoshop Document)* format yang menyimpan gambar dalam bentuk *layer*, termasuk teks, *mask*, *opacity*, *blend mode*, *channel* warna, *channel alpha*, *clipping paths*, dan *setting duotone*. Kepopuleran *photoshop* membuat format file ini digunakan secara luas, sehingga memaksa programmer program



penyunting gambar lainnya menambahkan kemampuan untuk membaca format PSD dalam perangkat lunak mereka.

- PSB adalah versi terbaru dari PSD yang didesain untuk file yang berukuran lebih dari 2 GB
- PDD adalah versi lain dari PSD yang hanya dapat mendukung fitur perangkat lunak Photoshop Deluxe.

### **2.5.6 3D Studio Max**

*3D Studio Max* merupakan sebuah perangkat lunak grafik vektor 3 dimensi dan animasi, ditulis oleh Autodesk Media & Entertainment. Sebelum namanya 3D Studio Max, dia dikenal sebagai Discreet and Kinetix. Perangkat lunak ini dikembangkan dari pendahulunya 3D Studio for DOS, tetapi untuk platform Win32. Kinetix kemudian bergabung dengan akuisisi terakhir Autodesk, Discreet Logic.

*3D Studio Max* adalah *software* untuk pembuatan animasi 3D. Sejak pertama kali dirilis, 3D Studio Max menjadi pemimpin aplikasi pembangunan animasi 3D. Sejak versi ke-empat, Discreet, produsen 3D Studio Max, berusaha untuk meluaskan area fungsinya sehingga dapat digunakan untuk membuat animasi bagi web atau film. Versi lainnya, yaitu versi 5, sudah mengarah kepada perluasan fungsi tersebut. Hal ini ditunjukkan dengan adanya pengembangan pada *polymodelling*, *mapping*, dan beberapa revisi pada *tools* untuk animasi. Namun dari fitur-fitur yang ada, fitur yang paling menarik dari 3D Studio Max versi 5 adalah *reactor*. *Reactor* ini terintegrasi dengan *interface* dari 3D Studio Max dan menyediakan *tool* untuk membuat simulasi. 3D Studio Max ini sering digunakan untuk membuat model-model rumah atau furnitur. Selain itu, banyak pula digunakan di dalam seni digital dan pembuatan game. Secara garis besar proses animasi 3D bisa dibagi 4 tahap yaitu:

#### **a. Modeling**

Tahap ini adalah pembuatan objek-objek yang dibutuhkan pada tahap animasi. Objek ini bisa berbentuk *primitive object* seperti *sphere* (bola), *cube* (kubus) sampai *complicated object* seperti sebuah karakter dan sebagainya. Ada

beberapa jenis materi obyek yang disesuaikan dengan kebutuhannya yaitu: *polygon*, *spline*, dan *metaclay*. *Polygon* adalah segitiga dan segiempat yang menentukan area dari permukaan sebuah karakter. Setiap *polygon* menentukan sebuah bidang datar dengan meletakkan sebuah jajaran *polygon* sehingga kita bisa menciptakan bentuk-bentuk permukaan. *Spline* adalah beberapa kumpulan *spline* yang membentuk sebuah lapisan kurva yang halus yang dinamakan *patch*. Sebuah *patch* menentukan area yang jauh lebih luas dan halus dari sebuah *polygon*. *Metaclay* dalam bentuk dasarnya, *metaball* berbentuk bola (*sphere*) yang bisa digabungkan satu sama lain sehingga membentuk bentuk organik obyek.

**b. Animating**

Pembuat animasi hanya menentukan/membuat *keyframe-keyframe* pada obyek yang akan digerakkan. Setelah proses *keyframing* dibuat, komputer akan menghitung dan membuat sendiri *inbetween* secara otomatis.

**c. Texturing**

Proses ini menentukan karakteristik sebuah materi obyek dari segi tekstur. Untuk materi sebuah obyek itu sendiri, kita bisa mengaplikasikan properti tertentu seperti *reflectivity*, *transparency*, dan *refraction*. *Texture* kemudian bisa digunakan untuk membuat berbagai variasi warna *pattern*, tingkat kehalusan/kekasaran sebuah lapisan obyek secara lebih detail.

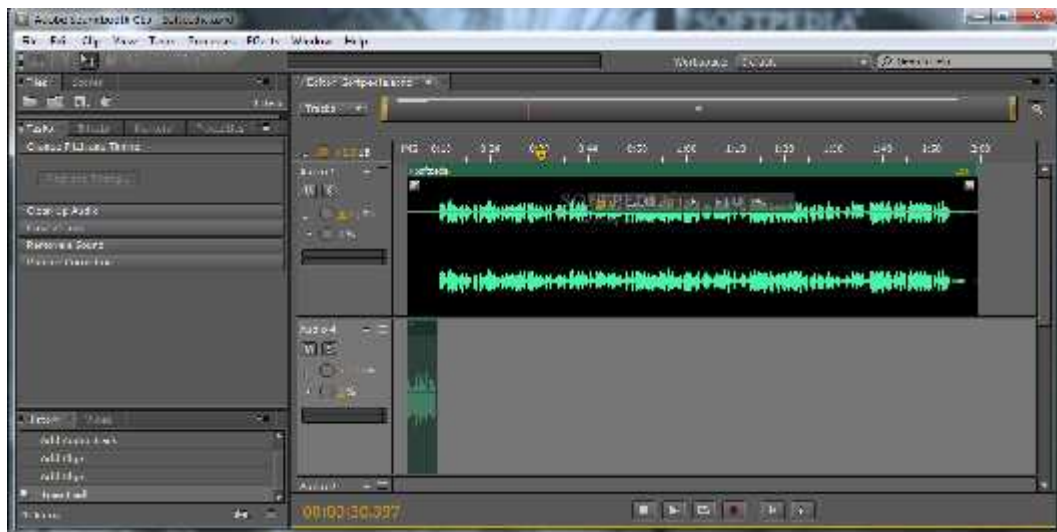
**d. Rendering**

*Rendering* adalah proses akhir dari keseluruhan proses animasi komputer. Dalam *rendering*, semua data-data yang sudah dimasukkan dalam proses *modelling*, *animasi*, *texturing*, pencahayaan dengan parameter tertentu akan diterjemahkan dalam sebuah bentuk keluaran.

### **2.5.7 Adobe Soundbooth**

Adobe Soundbooth adalah sebuah aplikasi editing audio yang kompatibel dengan file audio mandiri (misalnya WAV, AIFF, MP3, WMA) serta video file (misalnya AVI, MPEG-2, H.264, RealVideo, FLV). Adobe mempromosikan Soundbooth sebagai aplikasi pilihan bagi mereka yang bekerja dengan file video dan multimedia. Adobe Soundbooth merupakan sebuah aplikasi editing audio

pertama kali dirilis oleh Adobe pada tahun 2007. Adobe Soundbooth adalah audio sederhana namun cukup kuat mengedit aplikasi yang dirancang untuk mengedit audio dengan file video. Soundbooth memiliki keuntungan menjadi lebih mudah untuk belajar dan sederhana untuk digunakan untuk tugas-tugas umum tertentu. Tampilan awal pada *Adobe Soundbooth* dapat dilihat pada gambar 2.13.



**Gambar 2.2 Tampilan pada *Adobe Soundbooth***

## **2.6 *Rational Unified Process (RUP)***

*Rational Unified Process (RUP)* merupakan proses rekayasa perangkat lunak berorientasi objek. *RUP* menyediakan pendekatan disiplin tugas dari masing-masing tanggung jawab dari sistem organisasi. Tujuan dari *RUP* adalah untuk memastikan produksi perangkat lunak berkualitas tinggi yang memenuhi kebutuhan pengguna, jadwal dan prediksi anggaran (Rational Software, 1998).

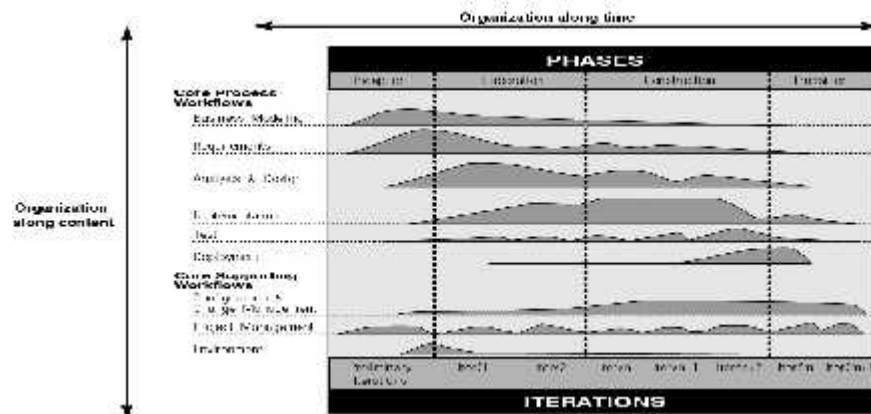
*RUP* memberikan pedoman kepada pengembang perangkat lunak pada pendekatan komersial yang efektif. *RUP* menyediakan pedoman, template dan alat pengajar yang diperlukan dan memberikan keuntungan dalam beberapa aspek diantaranya dalam praktek-praktek berikut ini.

1. Pengembangan perangkat lunak *iterative*.
2. Mengelola prasyarat.

3. Penggunaan komponen dasar asitektur.
4. Visual model perangkat lunak.
5. Verifikasi kualitas perangkat lunak
6. Kontrol pengembangan perangkat lunak.

### 2.6.1 Fase RUP

Fase dalam *RUP* sesuai dengan waktu pengerjaan proyek terbagi atas empat fase, yaitu *Inception*, *Elaboration*, *Construction* dan *Transition* seperti pada gambar 2.14 berikut (Rational Software, 1998).



**Gambar 2.3 Model grafik fase RUP**

#### 2.6.1.1 Fase Inception

Pada fase ini hal pertama yang harus dilakukan adalah membatasi ruang lingkup proyek. Untuk itu harus dilakukan indentifikasi semua entitas eksternal yang berinteraksi langsung dengan aktor dengan lebih dalam. Pada fase ini akan dilakukan identifikasi menggunakan *use case*, *class diagram*, *activity diagram* dan *sequence diagram* dengan minimal telah selesai 80%. Fase *inception* merupakan tonggak utama pada sebuah proyek yang merupakan tujuan penting siklus hidup proyek.

#### 2.6.1.2 Fase Elaboration

Tujuan dari fase *elaboration* ini adalah untuk menganalisa bidang masalah, dasar arsitektur, mengembangkan rencana proyek, dan menghilangkan resiko-resiko tertinggi dalam proyek. Adapun hasil dari fase *elaboration* adalah:

1. UML model telah selesai 100%, seluruh aktor telah teridentifikasi dan sebagian besar deskripsi diselesaikan
2. Arsitektur menu perangkat lunak.
3. Rancangan *prototype* yaitu berupa rancangan antar muka pengembangan keseluruhan perangkat lunak.

#### **2.6.1.3 Fase *Construction***

Pada fase ini produk dan fitur diuji akan mulai dibangun dan diimplementasikan sesuai dengan analisa dan perancangan pada fase *inception* dan *elaboration*. Hasil dari fase *construction* adalah produk siap untuk diberikan kepada *end-user* minimal terdiri dari beberapa kriteria berikut:

1. Produk perangkat lunak sudah dapat dijalankan pada perangkat device.
2. Penjelasan tentang rilis saat ini.

Pada akhir fase *construction* merupakan tonggak utama ketiga. Pada titik ini diputuskan perangkat lunak siap digunakan namun masih beresiko tinggi biasanya rilis ini disebut beta.

#### **2.6.1.4 Fase *Transition***

Tujuan dari tahap *Transition* ini adalah untuk mendapatkan hasil dari pengujian baik dari devicenya atau dari penggunaanya. Pengujian yang akan dilakukan adalah pengujian terhadap beberapa device dengan versi yang berbeda-beda, kemudian dilakukan pengujian terhadap pengguna. Setelah produk diuji coba, maka akan ditarik kesimpulan dari hasil pengujiannya.

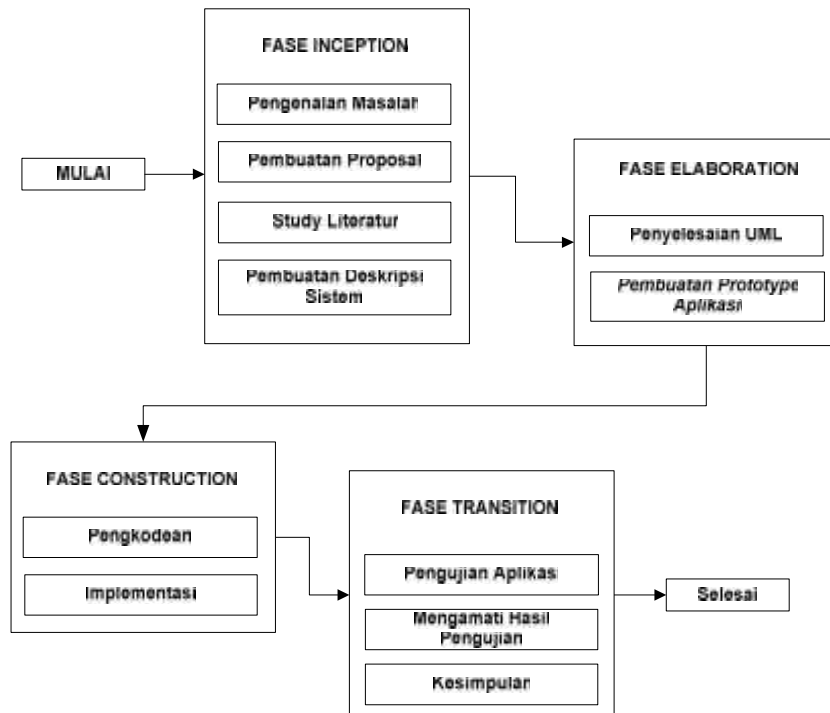
## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Tahapan Penelitian

Pada tahapan penelitian dalam pembuatan Tugas Akhir ini menggunakan metodologi *Rational Unified Process* (RUP). *RUP* merupakan salah satu metodologi pengembangan perangkat lunak berorientasi objek. Penelitian ini akan dibangun dengan menggunakan pemrograman *Android* yang *library*-nya dibangun dengan bahasa pemrograman *Java*. Penggunaan metode *RUP* disesuaikan dengan bahasa *Java* yang merupakan bahasa pemrograman berorientasi objek.

Tahapan tentang *RUP* sebelumnya telah sedikit dijelaskan pada bab landasan teori. Berikut adalah gambar dari tahapan penelitian dengan Metodologi *Rational Unified Process*.



**Gambar 3.1. Tahapan *Rational Unified Process* (RUP)**

### **3.2 Tahapan pengembangan *Rational Unified Process* (RUP)**

Berikut ini akan dijelaskan tahapan fase penelitian tentang aplikasi pembelajaran shalat yang akan dibuat dengan menggunakan aplikasi *Eclipse android* menggunakan bahasa *Java* sesuai dengan metode pengembangan *RUP*.

#### **3.2.1. Fase *Inception***

Fase *inception* merupakan fase tahapan identifikasi masalah yang ada pada aplikasi pembelajaran shalat yang akan dibangun sebagai penyelesaian Tugas Akhir ini. Pada fase ini akan diidentifikasi entitas yang berhubungan langsung dengan sistem melalui identifikasi *use case* beserta gambar *use case*. Berikut ini adalah hasil dari fase *inception* yang harus dicapai:

1. Setelah dilakukan pengenalan masalah, kemudian dilakukan identifikasi masalah meliputi cakupan masalah, kebutuhan penelitian dan identifikasi kendala.
2. Pembuatan proposal tentang Aplikasi shalat meliputi, latar belakang masalah, rumusan masalah, batasan masalah, dan tujuan.
3. Melakukan studi literatur terhadap teori yang dibutuhkan dalam implementasi aplikasi shalat. Literatur yang dikumpulkan antara lain adalah :
  - a. Mencari buku tata cara shalat, mulai dari gerakan sampai bacaan shalat yang sesuai dengan tuntunan Rasulullah SAW.
  - b. Mencari kebutuhan dari aplikasi yang akan dibangun, mulai dari tool yang digunakan, lingkungan pengembangan dan fitur-fitur yang akan dibangun.
4. Pembuatan deskripsi sistem dengan melakukan perancangan *use case*, *class diagram*, *activity diagram* dan *sequence diagram*.

#### **3.2.2. Fase *Elaboration***

Fase *elaboration* ini adalah fase pengembangan dari masalah yang telah diidentifikasi dan menganalisa terhadap aplikasi pembelajaran shalat.. Hasil yang harus dicapai pada fase *elaboration* adalah:

1. UML yang telah selesai setidaknya 80 % terhadap seluruh entitas yang terlibat dalam aplikasi.
2. Deskripsi dari rancangan arsitektur menu aplikasi shalat disesuaikan dengan kebutuhan aktor.
3. Rencana pengembangan keseluruhan program aplikasi shalat dengan *prototype* yang berupa rancangan antar muka keseluruhan aplikasi.

### **3.2.3. Fase *Construction***

Pada fase *construction* yaitu akan dilakukan pembangunan aplikasi pembelajaran shalat sesuai dengan rancangan dan deskripsi yang telah dibuat sebelumnya pada fase *inception* dan *elaboration*. Dari penjelasan di atas maka hasil yang harus dicapai pada proses *Construction* adalah :

1. Proses membangun aplikasi shalat dengan tools *Eclipse* untuk platform android menggunakan bahasa pemrograman *Java* telah diselesaikan.
2. Telah dilakukan implementasi aplikasi yang pertama dirilis menggunakan *device handphone* dengan Sistem operasi Android.

### **2.2.4 Fase *Transition***

Fase ini merupakan tahapan evaluasi aplikasi dengan melakukan pengujian terhadap beberapa device dengan beberapa versi yang berbeda-beda. Kemudian pengujian terhadap pengguna Pada tahapan ini akan terlihat kekurangan dari pada aplikasi yang dibuat dari jawaban pengguna terhadap aplikasi yang dibuat. Berikut hasil yang harus dicapai pada tahapan ini:

1. Melakukan pengujian dengan *pada device*, yaitu pengujian dilakukan Pada *Device Android* yang berbeda beda tidak hanya pada satu smartphone Android saja, apakah aplikasi nantinya berjalan dengan baik atau tidak.
2. Melakukan pengujian terhadap pengguna yaitu dengan menyebar kuisioner kepada pengguna, yaitu anak-anak smp kelas 1 (satu) antara umur 12 sampai 13 tahun.
3. Melakukan pengamatan terhadap hasil pengujian yang telah dilakukan.



4. Membuat kesimpulan terhadap hasil pengujian dan merangkum kekurangan aplikasi agar dapat diperbaiki dan dikembangkan dikemudian hari.

## **BAB IV**

### **ANALISA DAN PERANCANGAN**

Bab ini merupakan bagian dari fase *inception* dan fase *elaboration*, dimana akan dilakukan analisa sejalan dengan pembuatan deskripsi arsitektur yang dibutuhkan aplikasi dan pembuatan UML (*Unified Modelling Language*) sebagai bagian dari fase *inception*, dan dilakukan perancangan struktur menu aplikasi yang akan dibangun, dan perancangan antarmuka aplikasi yang akan dibangun sebagai bagian dari fase *elaboration*.

#### **4.1. Fase *Inception* dan Fase *Elaboration***

##### **4.1.1. Gambaran Umum Sistem**

Aplikasi shalat yang dibahas dalam penelitian ini adalah aplikasi yang akan dijalankan pada *smartphone* dengan sistem operasi android dan memiliki fungsi untuk mempermudah *user* dalam belajar shalat wajib. Aplikasi yang akan dibangun ini nantinya berbasis multimedia, dimana *user* tidak hanya dapat membaca dan melihat gambar, tetapi *user* juga dapat mendengarkan bacaan-bacaan doa dalam shalat dan menonton video shalat dalam bentuk video 3D.

Aplikasi shalat ini nantinya akan mempunyai empat menu utama. Dimana menu-menu itu terdiri dari yang pertama dalil tentang perintah shalat, baik dari Alqur'an maupun Hadits. Menu kedua merupakan menu persiapan shalat, sedangkan menu ketiga merupakan menu rukun shalat yang terdiri takbir sampai dengan salam. Menu yang terakhir yaitu panduan yang berisi tentang video 3D seluruh shalat lima waktu.

##### **4.1.2. Deskripsi Kebutuhan Aplikasi**

Untuk membangun sebuah aplikasi yang efisien maka kebutuhan dari aplikasi merupakan hal yang harus diperhatikan. Mengetahui kebutuhan aplikasi akan membantu dan memudahkan dalam pembangunan aplikasi itu sendiri.

#### 4.1.2.1. Aplikasi yang akan dibangun

Untuk kebutuhan aplikasi yang akan dibangun terdapat dua bagian yakni yang pertama sistem pada perangkat android dan yang kedua adalah multimedia.

##### a. Kebutuhan Sistem

Software yang dibutuhkan dalam membuat atau membangun sebuah aplikasi adalah:

1. Bahasa pemrograman yang digunakan adalah *Java Android*, untuk itu kita perlu menginstal java (disini penulis menggunakan *jdk-6u10-windows*) kedalam komputer atau laptop tempat kita bekerja nantinya.
2. Selanjutnya yang perlu kita instalasi adalah Android SDK, karena SDK inilah yang kita perlukan sebagai alat bantu dan API dalam mengembangkan aplikasi pada *platform* android menggunakan bahasa pemrograman java.
3. Kemudian Eclipse sebagai IDE (*Integrated Development Environment*) yang merupakan tempat kita membuat emulator *device* android yaitu dimana kita akan membuat *coding project android*, kemudian hasil pemrograman akan ditampilkan dan di uji coba.
4. Selanjutnya kita melakukan instalasi ADT atau *plugins eclipse*, dengan *plugins* inilah yang membuat eclipse dapat membuat *project* yang berbasis android.

##### b. Multimedia

Kebutuhan multimedia terdiri dari gambar, suara dan video.

1. Untuk gambar sendiri akan dilakukan pengeditan di *photoshop*, untuk itu kita perlu instalasi *photoshop* (disini penulis menggunakan *photoshop CS2*).
2. Selanjutnya instalasi *adobe soundbooth*, karena untuk suara bacaan dari doa-doa dalam shalat dilakukan pengeditan menggunakan *adobe soundbooth*.
3. Video akan dilakukan pertama kali pemodelan menggunakan 3D max dan kemudian dari model tersebut akan dibuat videonya, untuk itu kita perlu melakukan instalasi *3D Studio Max*.

#### **4.1.2.2. Analisa fungsional**

Analisa fungsional merupakan penjelasan mengenai menu-menu yang ada pada aplikasi shalat. Menu-menu tersebut antara lain:

1. Untuk tampilan awalnya, user akan menjumpai tiga menu, yaitu Shalat, About dan Exit
2. Menu shalat akan membawa user masuk ke dalam empat menu utama, yaitu Dalil, Persiapan, Rukun dan Panduan.
3. Didalam menu dalil, aplikasi akan menampilkan dalil perintah shalat dari Al-qur'an dan Hadits Nabi Muhammad SAW.
4. Untuk menu persiapan, aplikasi akan menampilkan info apa saja yang dilakukan sebelum melakukan shalat
5. Menu ketiga yaitu rukun, didalam menu ini aplikasi akan menampilkan urutan dari gerakan-gerakan shalat dan video dan bacaan-bacaan dalam shalat mulai dari takbiratul ikhram sampai salam.
6. Untuk menu panduan, aplikasi akan menampilkan video shalat lima waktu.

#### **4.1.3. Perancangan Sistem**

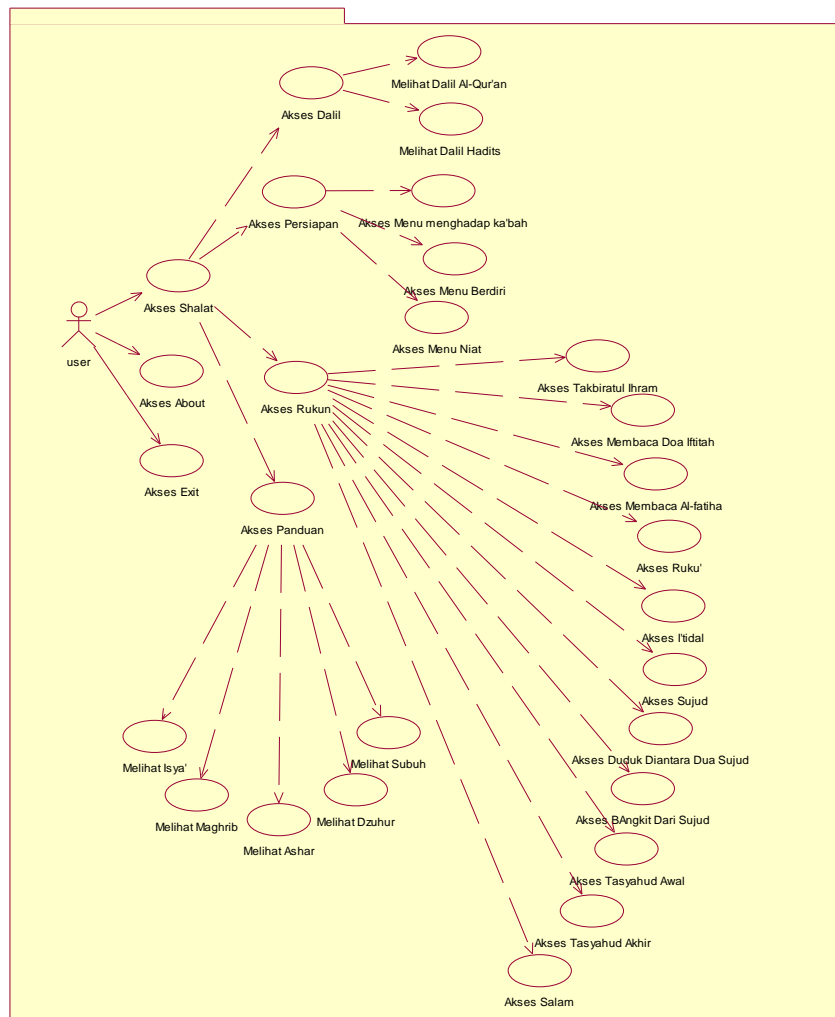
Setelah dilakukan beberapa tahapan dalam analisa sistem, maka dapat dilakukan beberapa perancangan aplikasi shalat pada sistem operasi Android. Perancangan-perancangan yang akan dijelaskan dalam laporan ini meliputi perancangan model dalam bentuk UML (*Unified Modeling Language*) yang terdiri dari *Usecase Diagram*, *Class Diagram*, *Activity Diagram*, dan *Sequence Diagram*. Selain itu juga ada perancangan hubungan antar *class* di Android dengan modul penghubung dan perancangan sistem yang terdiri dari struktur menu dan perancangan *interface*.

##### **4.1.3.1. Pemodelan UML (*Unified Modeling Language*)**

Analisa perancangan pada sistem ini menggunakan UML, yaitu terdiri dari *Usecase Diagram*, *Class Diagram*, *Activity Diagram*, dan *Sequence Diagram*.

#### 4.1.3.1.1. Usecase Diagram

*Usecase diagram* merupakan suatu aktivitas yang menggambarkan urutan interaksi antar satu atau lebih aktor dan sistem. *Usecase* yang akan dirancang yaitu *usecase diagram* untuk pengaksesan melalui perangkat Android. Gambar 4.1 dibawah ini menjelaskan aliran *usecase diagram* pengaksesan melalui perangkat Android.



**Gambar 4.1. Usecase diagram**

Dari gambar 4.1 dapat dilihat sistem ini terdiri dari 1 aktor dan 3 *usecase*. Untuk lebih jelasnya, Spesifikasi dari *usecase diagram* (pengaksesan melalui perangkat Android) dapat dilihat pada tabel dibawah ini.

**Tabel 4.1. Spesifikasi Akses Dalil**

|                       |  |
|-----------------------|--|
| <i>Actor</i>          | <i>User</i>  |
| <i>Use Case</i>       | Akses Dalil  |
| <i>Description</i>    | User mengakses menu dalil untuk melihat menu dalil               |
| <i>Pre Condition</i>  | Sistem menampilkan halaman menu awal                             |
| <i>Post Condition</i> | Aktor berhasil melakukan akses ke menu dalil untuk melihat dalil |

**Tabel 4.2. Spesifikasi Akses Persiapan**

|                       |   |
|-----------------------|---|
| <i>Actor</i>          | <i>User</i>   |
| <i>Use Case</i>       | Akses Persiapan   |
| <i>Description</i>    | User mengakses menu persiapan untuk melihat isi menu persiapan                            |
| <i>Pre Condition</i>  | Sistem menampilkan halaman menu awal  |
| <i>Post Condition</i> | Aktor berhasil melakukan akses ke menu persiapan untuk melihat isi halaman menu persiapan |

**Tabel 4.3. Spesifikasi Akses Rukun**

|                       |   |
|-----------------------|---|
| <i>Actor</i>          | <i>User</i>   |
| <i>Use Case</i>       | Akses Rukun   |
| <i>Description</i>    | User mengakses menu rukun untuk melihat menu rukun                            |
| <i>Pre Condition</i>  | Sistem menampilkan halaman menu awal  |
| <i>Post Condition</i> | Aktor berhasil melakukan akses ke menu rukun untuk melihat halaman menu rukun |

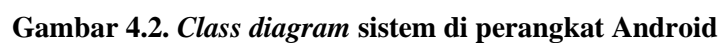
**Tabel 4.4. Spesifikasi Akses Panduan**

|                       |   |
|-----------------------|---|
| <i>Actor</i>          | <i>User</i>   |
| <i>Use Case</i>       | Akses Panduan   |
| <i>Description</i>    | User mengakses menu panduan untuk melihat menu panduan                                |
| <i>Pre Condition</i>  | Sistem menampilkan halaman menu awal  |
| <i>Post Condition</i> | Aktor berhasil melakukan akses ke menu panduan untuk melihat isi halaman menu panduan |

Untuk spesifikasi dari *usecase diagram* lainnya dapat dilihat pada lampiran B.

#### **4.1.3.1.2. Class Diagram**

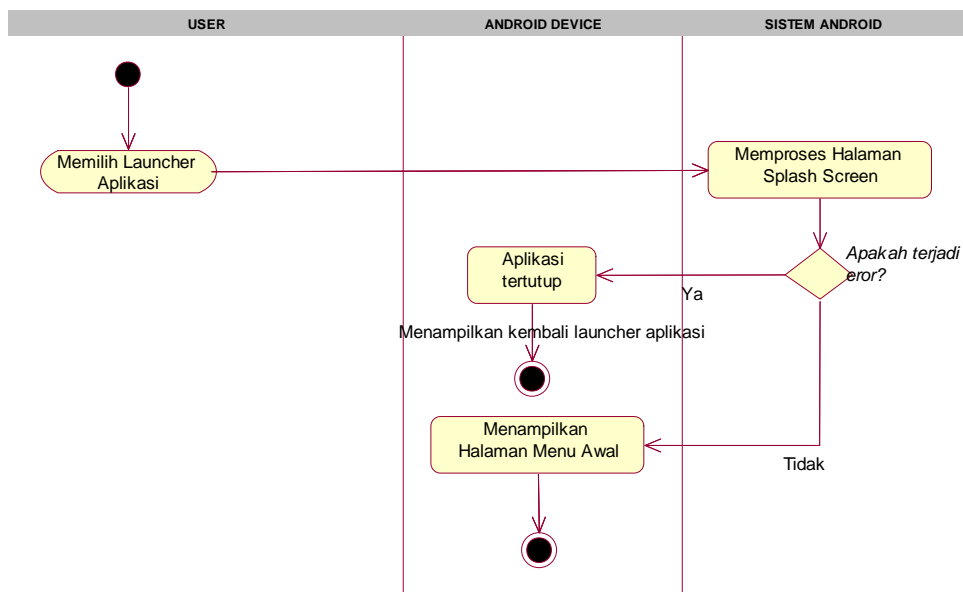
*Class Diagram* menggambarkan struktur dan deskripsi *class*, *package*, dan objek yang saling terhubung. *Class diagram* yang dijelaskan pada analisa ini adalah *class diagram* sistem yang terpasang pada perangkat Android. Gambar 4.2 dibawah ini menjelaskan *class diagram* sistem secara umum yang terpasang pada perangkat Android dan untuk class diagram aplikasi dapat dilihat pada Lampiran B.



*Activity diagram* merupakan alur kerja pada setiap *usecase*. *Activity diagram* pada analisa ini mencakup *activity diagram* setiap *usecase*.

## IV - 7

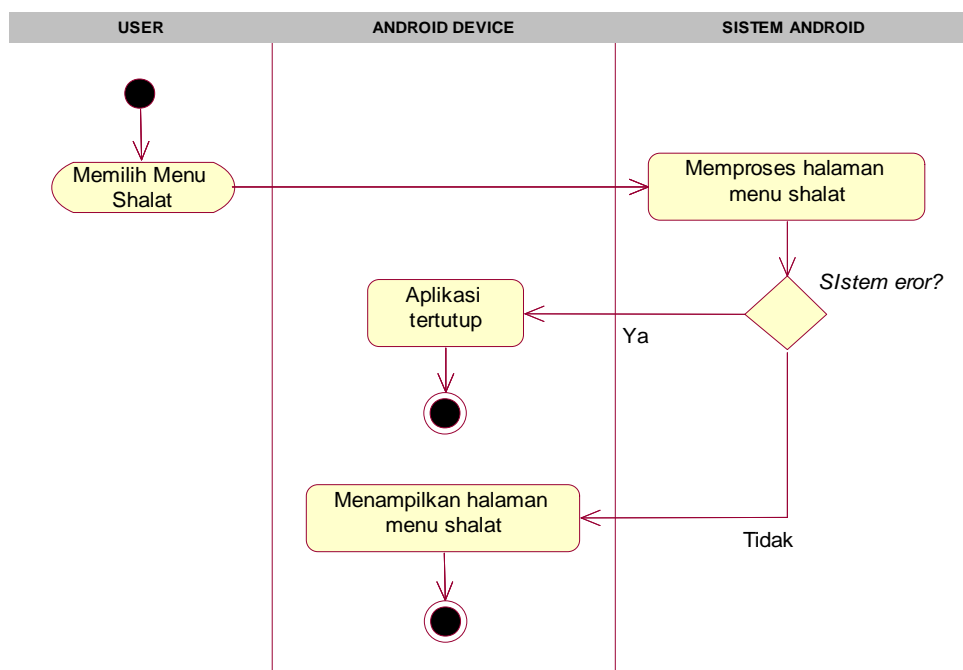




**Gambar 4.3. Activity diagram buka aplikasi**

Pada bagian pertama *activity diagram* seperti terlihat pada gambar 4.3 saat membuka aplikasi maka sistem akan menampilkan menu awal aplikasi yang terdiri dari tiga pilihan menu.

#### B. Activity Diagram Menu Shalat

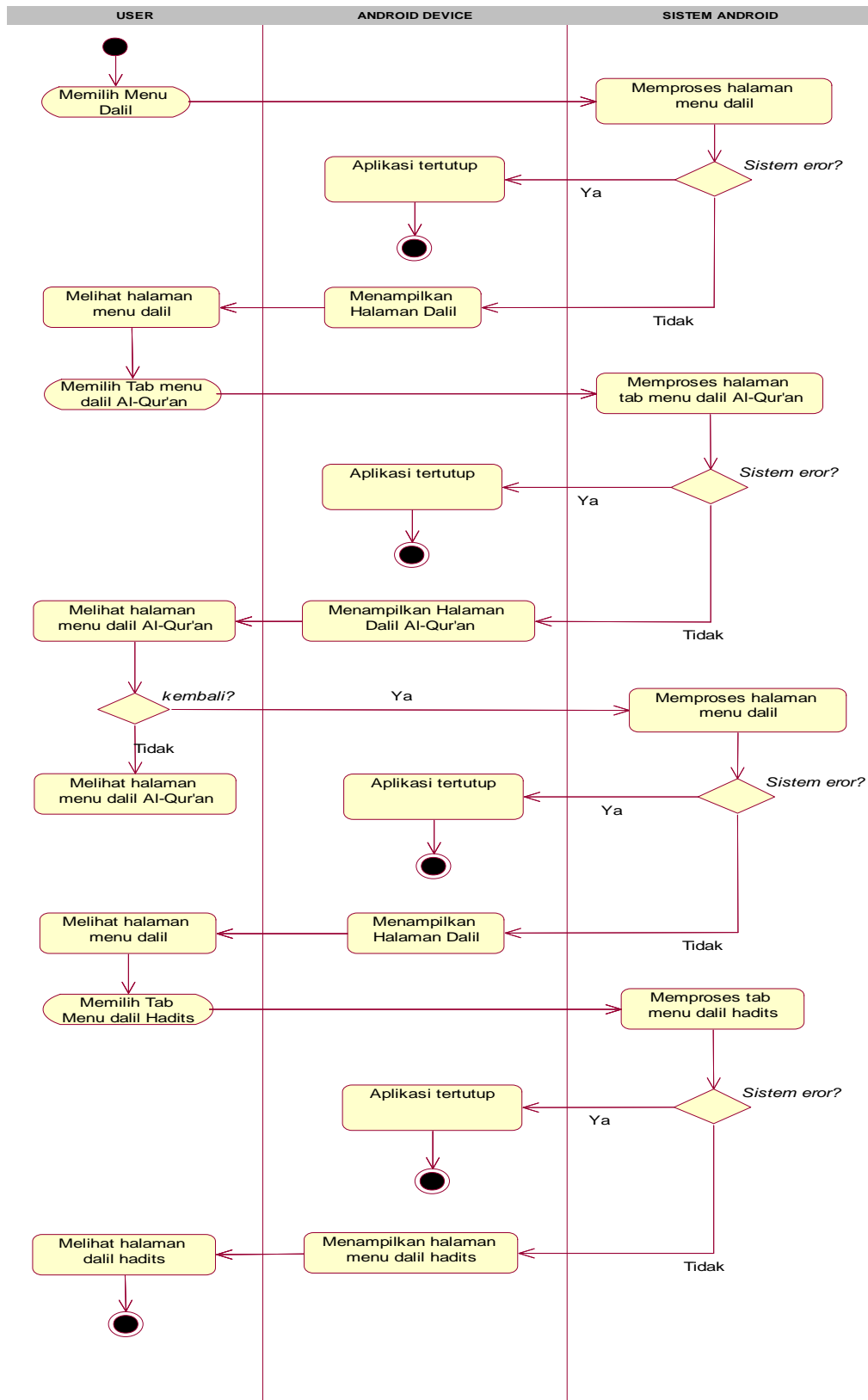


**Gambar 4.4. Activity diagram menu shalat**

*Activity diagram* menu shalat alur proses lanjutan dari *activity diagram* buka aplikasi. Alur kerja dari proses ini adalah setelah *user* masuk aplikasi kemudian *user* akan melihat tiga menu awal yaitu shalat, about, dan exit. Pada diagram gambar 4.4 *user* memilih menu shalat maka akan sistem akan menampilkan empat menu utama yang terdapat didalam menu shalat.

#### C. *Activity Diagram* Menu Dalil

*Activity* ini merupakan alur proses setelah membuka menu shalat. Didalam menu shalat terdapat empat menu utama yang pertama yaitu menu dalil, yang kedua menu persiapan, yang ketiga menu rukun dan yang keempat adalah menu panduan. *Activity diagram* menu dalil dapat kita lihat pada gambar 4.5 dibawah ini.



Gambar 4.5. Activity diagram menu dalil

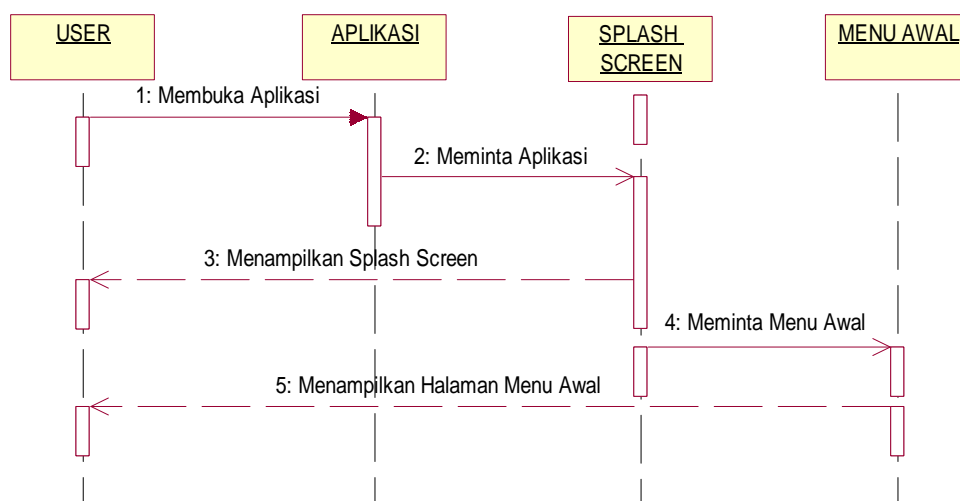
Didalam activity diagram diatas, dapat dilihat bahawa menu dalil terdapat dua menu tab yaitu tab menu Al-Qur'an dan menu Hadist. Tab menu Al-Qur'an menjelaskan tentang perintah shalat dari ayat Al-Qur'an. Tab menu Hadist menjelaskan peerintah shalat dari sunah Rasulullah SAW, dan untuk activity diagram yang lain dapat dilihat pada lampiran B.

#### 4.1.3.1.4. *Sequence Diagram*

Sequence Diagram adalah representasi dari interaksi-interaksi objek yang berjalan pada sistem. Dengan menggunakan *sequence diagram* kita dapat melihat bagaimana objek-objek bekerja. *Sequence diagram* dapat menampilkan bagaimana sistem merespon setiap kejadian atau permintaan dari user, dapat mempertahankan integritas internal, bagaimana data dipindah ke user interface dan bagaimana objek-objek diciptakan dan dimanipulasi.

Setiap sistem memiliki proses dan setiap proses memiliki dua kriteria, yaitu proses sederhana dan kompleks. Dengan demikian tidak seluruh proses pada sistem akan ditampilkan pada sequence diagram, melainkan hanya garis besarnya saja.

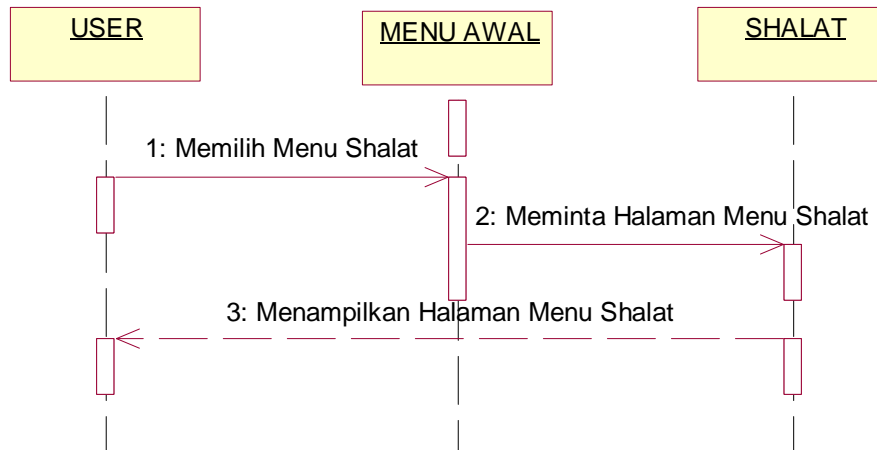
##### A. *Sequence Diagram* Buka Aplikasi



**Gambar 4.6 *Sequence Diagram* buka aplikasi**

Dari Gambar diatas terlihat user memilih aplikasi shalat kemudian tampil *splash screen* dan sistem meminta menu awal yang kemudian sistem menampilkan menu awal.

#### B. *Sequence Diagram* Shalat



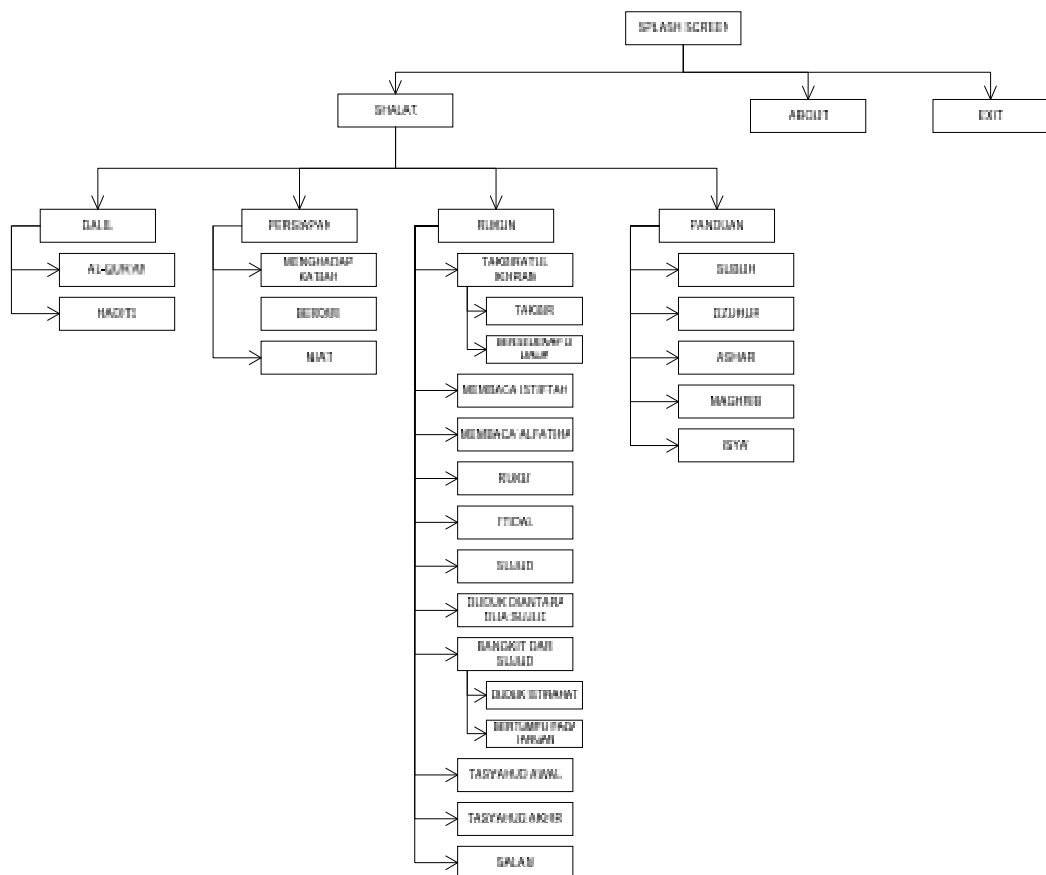
**Gambar 4.7** *Sequence Diagram* Shalat

Dari Sequence Diagram Shalat diatas, user memilih menu shalat maka sistem akan menampilkan halaman menu shalat kepada user.

#### 4.2. Fase *Elaboration*

##### 4.2.1. Perancangan Struktur Menu Sistem

Rancangan struktur menu merupakan tahapan untuk merancang bagaimana struktur menu yang akan dibangun. Berikut struktur menu dari sistem yang akan dibangun dapat dilihat pada gambar 4.8 di bawah ini.



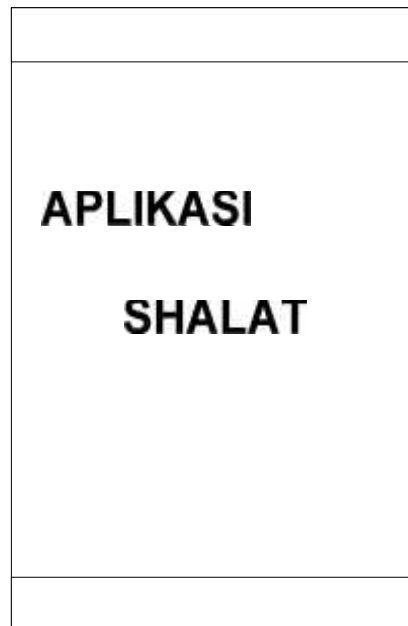
**Gambar 4.8. Rancangan Struktur menu**

#### 4.2.2. Perancangan *Interface*

Berikut adalah contoh tampilan interface aplikasi shalat yang berjalan pada sistem operasi android.

##### A. Halaman *Splash Screen*

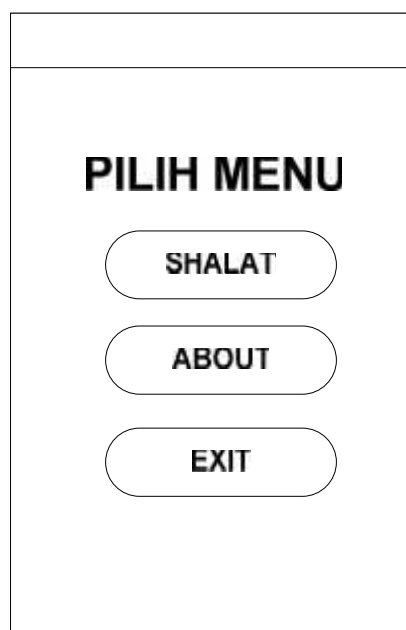
Halaman *splash screen* aplikasi ini tampil saat pertama kali *user* membuka aplikasi shalat pada *device* android yang dapat dilihat seperti gambar 4.9 berikut :



**Gambar 4.9 *Splash Screen* Aplikasi Shalat**

#### **B. Menu Awal**

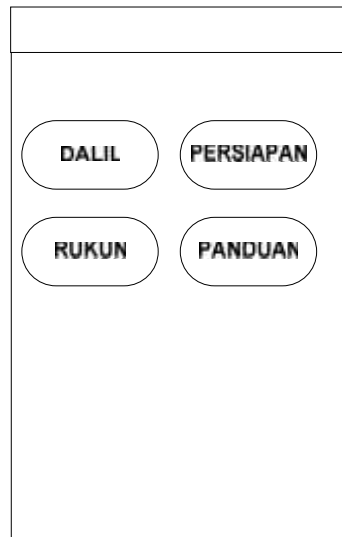
Menu awal akan muncul setelah *splash screen* ditampilkan. Pada *interface* menu awal ini , user akan melihat tiga pilihan menu aplikasi shalat seperti yang dapat dilihat pada gambar 4.10 berikut ini.



**Gambar 4.10 Tampilan Menu Awal**

### C. Menu Shalat

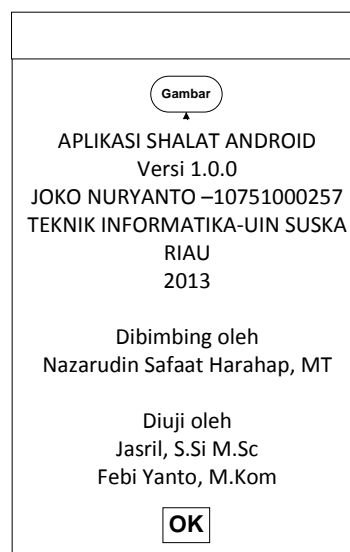
Jika *user* memilih menu shalat maka akan tampil empat menu utama dari aplikasi shalat ini, tampilan menu ini dapat dilihat pada gambar 4.11 sebagai berikut :



**Gambar 4.11 Tampilan Menu Shalat**

### D. Menu About

Halaman menu *about* dapat dilihat pada gambar 4.12. Halaman ini menampilkan informasi tentang aplikasi shalat. Untuk lanjutan perancangan *interface* dapat dilihat pada lampiran B.

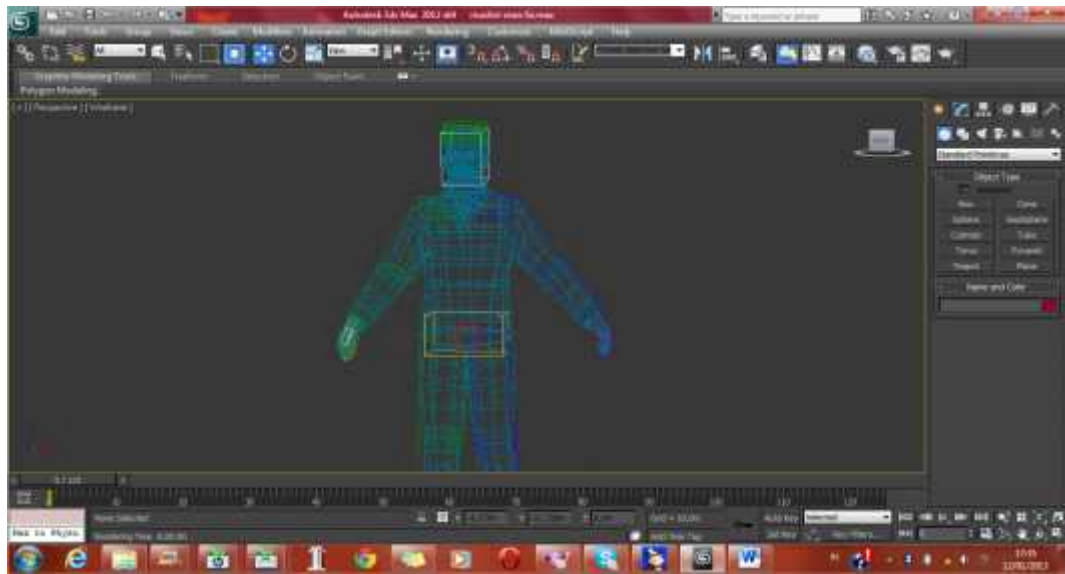


**Gambar 4.12 Tampilan Halaman Menu About**



#### 4.2.3. Perancangan Animasi 3D

Berikut merupakan perancangan model dari animasi orang shalat dalam 3D max, pemodelan pertama dengan menggunakan garis-garis atau *spline* yang dapat dilihat pada gambar 4.13.



**Gambar 4.13 Tampak depan pemodelan masih dengan garis-garis**

Dan berikut model objek yang telah dilakukan *animating*, *texturing* dan *rendering* seperti yang terlihat pada gambar 4.14 berikut ini.



**Gambar 4.14 Tampak Depan Pemodelan akhir**

## **BAB V**

### **IMPLEMENTASI DAN PENGUJIAN**

Bab ini merupakan bagian dari fase *construction* dan fase *transition*, dimana telah dilakukan pengkodean aplikasi, dan akan dilakukan implementasi aplikasi dan pengujian fungsi-fungsi aplikasi sebagai bagian dari fase *costruction*, dan akan dilakukan pengujian aplikasi beta terhadap pengguna, kemudian akan dilakukan pengamatan dari hasil pengujian tersebut untuk mengetahui kekurangan aplikasi dan kemudian dilakukan pengambilan kesimpulan sebagai bagian dari fase *transition*.

#### **5.1. Fase Construction**

##### **5.1.1. Implementasi**

Tahapan implementasi merupakan tahapan dimana aplikasi yang telah dirancang, dianalisa, dan dibangun, lalu diuji kelayakannya untuk selanjutnya dioperasikan sebagaimana mestinya sesuai dengan fungsi dan kelayakannya. Berikut ini akan dijelaskan tentang implementasi dari analisis dan perancangan yang telah dilakukan terhadap aplikasi shalat pada sistem operasi Android ini.

##### **5.1.1.1. Lingkungan Implementasi**

Lingkungan yang dibutuhkan untuk mengembangkan aplikasi ini antara lain berupa komponen perangkat keras dan perangkat lunak.

###### **1. Perangkat keras**

|              |  |
|--------------|--|
| Nama         | : Acer Aspire AM1900   |
| Processor    | : <i>Intel<sup>(R)</sup> Pentium<sup>(R)</sup> processor E5800</i> |
| Memori (RAM) | : 3 GB DDR3  |
| Hardisk      | : 250 GB   |

###### **2. Perangkat Lunak**

Sistem Operasi : *Windows 7 Ultimate 32-bit Operating System*

Bahasa Pemrograman : Java Android

Tools Pengembangan : Eclipse

. : *Java Development Kit*

: *Android SDK, ADT*

: *Android Virtual Device*

: *Adobe Soundbooth*

: *3D Max*

: *Photoshop*

Untuk lingkungan implementasi aplikasi pada device androidnya yaitu:

- a. Perangkat keras : *Smartphone Android “ Smartfren AD683G”*
- b. Perangkat lunak : *Sistem operasi Android 4.0.4*

### 5.1.2. Implementasi Kelas

Kelas-kelas yang telah dirancang diimplementasikan dengan menggunakan bahasa pemrograman java. Pada table 5.1 dapat dilihat daftar implementasi kelas-kelas yang ada pada aplikasi shalat.

**Tabel 5.1 Implementasi Kelas**

| Nama Kelas | Nama File   | Implementasi  |
|------------|-------------|---|
| splash     | Splash.java | <pre>public void run(){<br/><br/>                try{<br/><br/>                    super.run();<br/><br/>                    while<br/>(wait &lt; welcomeScren){<br/>                        sleep(100);<br/>                        wait+=100;<br/>                    }<br/><br/>                }catch(Exception e){<br/>                    System.out.println("Exc=" +e);<br/>                }finally{<br/><br/>                    startActivity(new<br/>Intent(splash.this,</pre> |

|               |                   |  |
|---------------|-------------------|--|
|               |                   | <pre> MainActivity.class));          finish();     }  };  welcomeThread.start(); }  } </pre>   |
| Main Activity | MainActivity.java | <pre> public void onCreate(Bundle savedInstanceState) {      super.onCreate(savedInstanceState);      setContentView(R.layout.activity_main);      form = new MenuForm(this);      form.getButtonBuatKontakBaru().setOnClickListener(this);      form.getBtnInfo().setOnClickListener(this);      form.getBtnExit().setOnClickListener(this); }  public void onClick(View v){     if(v == form.getButtonBuatKontakBaru()){         Intent intent = new Intent(this, BuatActivity.class);         startActivity(intent);      }else if(v == form.getBtnInfo()){         Intent intent = new Intent(this, tentang_aplikasi.class);         startActivity(intent);     }else if(v == form.getBtnExit()){         finish();     } } } </pre> |
| Buat Activity | BuatActivity.java | <pre> public void onClick(View v){      if(v == form.getButtonDalil()){         Intent intent = new Intent(this, Tab_Dalil.class);         startActivity(intent);      }else if(v == </pre>  |

|           |                |   |
|-----------|----------------|---|
|           |                | <pre> form.getButtonDetil()){     Intent intent = new Intent(<b>this</b>, RukunSolatActivity.class);     startActivity(intent);      }else if(v == form.getButtonPandu()){     Intent intent = new Intent(<b>this</b>, PanduanActivity.class);     startActivity(intent);      }else if(v == form.getButtonSyarat()){     Intent intent = new Intent(<b>this</b>, Persiapan.class);     startActivity(intent);     }     } } </pre>   |
| Tab Dalil | Tab_dalil.java | <pre> <b>protected void</b> onCreate(Bundle savedInstanceState) {     // <b>TODO</b> Auto-generated method stub      <b>super</b>.onCreate(savedInstanceState );      setContentView(<u>R</u>.layout.dalil_sol at);      Resources res = getResources();     TabHost tabHost = getTabHost();     TabHost.TabSpec spec;     Intent intent;      intent = <b>new</b> Intent().setClass(<b>this</b>, Dalil_alquran_Activity.class);     spec = tabHost.newTabSpec("Al- quran").setIndicator("Al- quran",res.getDrawable(<u>R</u>.drawable.dalil _tab_alquran)).setContent(intent);     tabHost.addTab(spec);      intent = <b>new</b> Intent().setClass(<b>this</b>, Dalil_hadis_Activity.class);     spec = tabHost.newTabSpec("Hadis").setIndicato r("Hadis",res.getDrawable(<u>R</u>.drawable.da lil_tab_hadis)).setContent(intent); </pre> |

|                      |                         |   |
|----------------------|-------------------------|---|
|                      |                         | <pre> tabHost.addTab(spec);  tabHost.setCurrentTab(2);      } } </pre>  |
| Rukun Solat Activity | RukunSolatActivity.java | <pre> private void tampilkanPilihan(String pilihan){     try{         Intent i = null;         if (pilihan.equalsIgnoreCase("Takbiratul Ihram")){             i = new Intent(this, Takbiratul_listmenu.class);          }else if (pilihan.equalsIgnoreCase("Membaca do'a istiftah")){             i= new Intent(this, Istiftah.class);          }else if (pilihan.equalsIgnoreCase("Membaca Al-fatiha")){             i= new Intent(this, Alfatiha.class);          }else if (pilihan.equalsIgnoreCase("Ruku')){             i = new Intent(this, rukuk.class);          }else if (pilihan.equalsIgnoreCase("sujud")){             i = new Intent(this, sujud.class);          }else if (pilihan.equalsIgnoreCase("I'tidal")){             i = new Intent(this, iktidal.class);          }else if (pilihan.equalsIgnoreCase("Duduk Diantara dua Sujud")){             i = new Intent(this, bangkitSujud.class);          }else if (pilihan.equalsIgnoreCase("Bangkit dari sujud")){             i = new Intent(this, Bangkitsujud_list.class); </pre> |

|                     |                          |   |
|---------------------|--------------------------|---|
|                     |                          | <pre>                 }else if                 (pilihan.equalsIgnoreCase("Tasyahhud                 Awal")){                     i= new Intent(this,                 dudukAwal.class);                  }else if                 (pilihan.equalsIgnoreCase("Tasyahhud                 Akhir")){                     i = new Intent(this,                 DudukAkhir.class);                  }else if                 (pilihan.equalsIgnoreCase("Salam")){                     i = new Intent(this,                 Salam.class);                  }startActivity(i);                  }catch(Exception e){                  }             }         }     } </pre>   |
| Panduan<br>Activity | PanduanActiv<br>ity.java | <pre> private void tampilkanPilihan(String pilihan) {     // TODO Auto-generated     method stub     try{         Intent i = null;         if         (pilihan.equalsIgnoreCase("Subuh")){             i = new             Intent(this, SubuhActivity.class);         }else if         (pilihan.equalsIgnoreCase("Zuhur")){             i = new             Intent(this, ZuhurActivity.class);         }else if         (pilihan.equalsIgnoreCase("Asyar")){             i = new             Intent(this, ZuhurActivity.class);         }else if         (pilihan.equalsIgnoreCase("Magrib")){             i = new             Intent(this, MagribActivity.class);         }else if         (pilihan.equalsIgnoreCase("Isya")){             i = new             Intent(this, ZuhurActivity.class);         }startActivity(i);          }catch(Exception e){          }     } </pre> |

|           |                |  |
|-----------|----------------|--|
|           |                | <pre>         }     } </pre>   |
| Persiapan | Persiapan.java | <pre> <b>private void</b> tampilkanPilihan(String pilihan) {     // <b>TODO</b> Auto-generated method stub     <b>try</b>{         Intent i = <b>null</b>;          <b>if</b>(pilihan.equalsIgnoreCase("Meng hadap Ka'bah")){             i = <b>new</b> Intent(<b>this</b>, MenghadapKiblat.<b>class</b>);         }<b>else if</b> (pilihan.equalsIgnoreCase("Berdiri")){             i = <b>new</b> Intent(<b>this</b>, Berdiri.<b>class</b>);         }<b>else if</b> (pilihan.equalsIgnoreCase("Niat")){             i = <b>new</b> Intent(<b>this</b>, Niat.<b>class</b>);         }         startActivity(i);      }<b>catch</b>(Exception e){      } } </pre> |

### 5.1.3. Hasil Implementasi Aplikasi pada Device Android

Sub bab ini berisi gambar-gambar hasil implementasi antar muka dari perangkat lunak yang telah dibangun. Perangkat lunak ini merupakan yang pertama rilis. Gambar hasil implementasi tersebut merupakan gambar dari device android Smartfren yaitu:

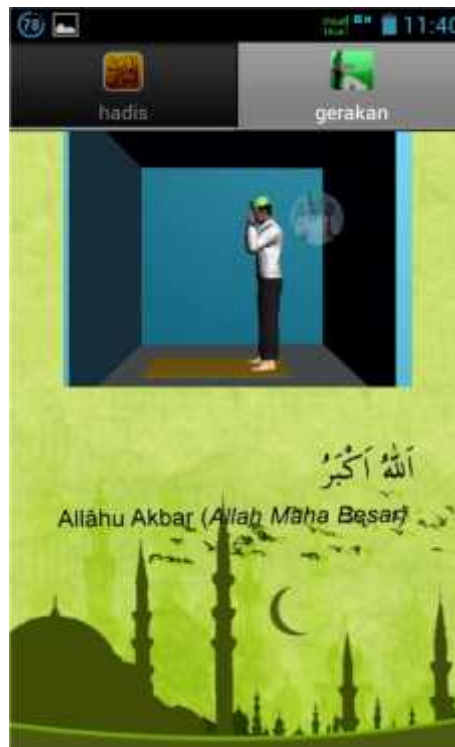




Gambar 5.1 Implementasi Antar Muka Splash Screen



Gambar 5.2 Implementasi Antar Muka Menu Utama



**Gambar 5.3 Implementasi Menu Takbir Tab Gerakan**



**Gambar 5.4 Implementasi Menu Rukuk Tab Gerakan**

Untuk implementasi antar muka selanjutnya dapat dilihat pada lampiran D.

## **5.2. Fase *Transition***

### **5.2.1. Pengujian Aplikasi Shalat Android**

Pengujian yang dilakukan bertujuan untuk mengetahui proses hasil dari aplikasi, yaitu memperlihatkan aplikasi shalat yang telah dibangun bisa diakses melalui berbagai perangkat Android dan berbagai versi Android. Pada tabel 5.2 di bawah ini menjelaskan pengujian akses ke aplikasi shalat menggunakan beberapa perangkat bersistem operasi Android, sedangkan pada table 5.3 menjelaskan pengujian interface dan menu.

**Tabel 5.2 Pengujian aplikasi shalat dari *device* Android**

| No | Device Android |  | Pengujian             | Hasil          | Ket                        |
|----|----------------|--|-----------------------|----------------|----------------------------|
| 1. | Nama           | HTC Hero   | Berjalan pada device  | Tidak Berjalan | Spesifikasi tidak memenuhi |
|    | Versi OS       | 1,5 (cupcake)  | Tampilan Interface    | Tidak ada      |                            |
|    | Memory         | 288 MB RAM<br>512 MB ROM                                 | Suara                 | Tidak ada      |                            |
|    | CPU            | QUALCOM 528 MHz<br>processor                             | Tampilan Gambar Video | Tidak ada      |                            |
|    | Layar          | 3.1 inch   |                       |                |                            |
| 2. | Nama           | HUAWEI U8800   | Berjalan pada device  | Tidak Berjalan | Spesifikasi tidak memenuhi |
|    | Versi OS       | 2,2 (froyo)  | Tampilan Interface    | Tidak ada      |                            |
|    | Memory         | 512 MB RAM<br>2 Gb Storage                               | Suara                 | Tidak ada      |                            |
|    | CPU            | QUALCOM 800 MHz<br>Scorpion processor,<br>Adreno 205 GPU | Tampilan Gambar Video | Tidak ada      |                            |
|    | Layar          | 3.8 inch   |                       |                |                            |

|    |                 |                             |                              |                           |   |
|----|-----------------|-----------------------------|------------------------------|---------------------------|---|
| 3. | <b>Nama</b>     | Samsung Galaxy Ace          | <b>Berjalan pada device</b>  | Berjalan                  | - |
|    | <b>Versi OS</b> | 2,3 (Gingerbread)           | <b>Tampilan Interface</b>    | Sesuai dengan perancangan |   |
|    | <b>Memory</b>   | 512MB RAM<br>3 Gb Storage   | <b>Suara</b>                 | Jelas                     |   |
|    | <b>CPU</b>      | QUALCOM 832 MHz processor   | <b>Tampilan Gambar Video</b> | Jelas, kecil              |   |
|    | <b>Layar</b>    | 3,5 inch                    |                              |                           |   |
| 4. | <b>Nama</b>     | Samsung Galaxy Tab P 7300   | <b>Berjalan pada device</b>  | Berjalan                  | - |
|    | <b>Versi OS</b> | 3,2 (Honeycomb)             | <b>Tampilan Interface</b>    | Sesuai dengan perancangan |   |
|    | <b>Memory</b>   | 16 GB storage,<br>1GB RAM   | <b>Suara</b>                 | Jelas, besar              |   |
|    | <b>CPU</b>      | Dual-core 1 GHz Cortex-A9   | <b>Tampilan Gambar Video</b> | Jelas, besar              |   |
|    | <b>Layar</b>    | 8,9 inch                    |                              |                           |   |
| 5. | <b>Nama</b>     | Smartfren Andromax-i AD683G | <b>Berjalan pada device</b>  | Berjalan                  | - |
|    | <b>Versi OS</b> | 4.0.4 (Ice Cream Sandwich)  | <b>Tampilan Interface</b>    | Sesuai dengan perancangan |   |

|           |                 |  |                              |                              |   |
|-----------|-----------------|--|------------------------------|------------------------------|---|
|           | <b>Memory</b>   | 4 GB, 512 MB RAM   | <b>Suara</b>                 | jelas                        |   |
|           | <b>CPU</b>      | Qualcomm MSM8625<br>Dual Core 1GHz, GPU:<br>Adreno 203       | <b>Tampilan Gambar Video</b> | Jelas                        |   |
|           | <b>Layar</b>    | 4.0 inch   |                              |                              |   |
| <b>6.</b> | <b>Nama</b>     | Smartfren Andromax-i<br>New AD796G                           | <b>Berjalan pada device</b>  | Berjalan                     | - |
|           | <b>Versi OS</b> | 4.1.2 (Jelly Bean)   | <b>Tampilan Interface</b>    | Sesuai dengan<br>perancangan |   |
|           | <b>Memory</b>   | 4 GB, 512 MB RAM   | <b>Suara</b>                 | jelas                        |   |
|           | <b>CPU</b>      | Qualcomm Snapdragon<br>Dual Core 1,2 GHz,<br>GPU: Adreno 203 | <b>Tampilan Gambar Video</b> | Jelas                        |   |
|           | <b>Layar</b>    | 4.0 inch   |                              |                              |   |

**Tabel 5.3 Pengujian *Interface* dan menu aplikasi shalat dari Device Android**

| <b>No.</b> | <b>Deskripsi</b>                 | <b>Kondisi Awal</b>                 | <b>Prosedur Pengujian</b>         | <b>Keluaran</b>   | <b>Hasil Evaluasi</b>                                       | <b>Hasil yang didapat</b>   | <b>Kesimpulan</b>        |
|------------|----------------------------------|-------------------------------------|-----------------------------------|---|---|---|--------------------------|
| 1.         | Pengujian Menu Launcher Aplikasi | Tampilan layar device               | Klik Launcher Aplikasi pada layar | berhasil diproses, menampilkan splash screen aplikasi shalat, kemudian menampilkan halaman menu aplikasi shalat | berhasil diproses, tampilan sesuai dengan hasil perancangan | berhasil diproses, tampilan halaman menu awal aplikasi sesuai perancangan                                     | Hasil Sesuai Perancangan |
| 2.         | Pengujian Menu Shalat            | Tampilan Halaman Menu Awal aplikasi | Pilih Menu Shalat                 | Berhasil diproses, menampilkan halaman 4 menu utama   | berhasil diproses, tampilan sesuai dengan hasil perancangan | Berhasil diproses, tampilan halaman 4 menu utama sesuai perancangan   | Hasil Sesuai Perancangan |
| 3.         | Pengujian Menu Dalil             | Tampilan Halaman 4 menu utama       | Pilih menu Dalil                  | Berhasil diproses, menampilkan halaman menu dalil   | berhasil diproses, tampilan sesuai dengan hasil perancangan | Berhasil diproses, tampilan menu dalil sesuai perancangan, menampilkan tab menu Al-qur'an dan tab menu hadist | Hasil Sesuai Perancangan |

Lanjutan table pengujian interface dapat dilihat pada lampiran E.

### 5.2.2. Pengujian Kepada Pengguna

Pengujian ini digunakan untuk menilai aplikasi yang berhubungan dengan tampilan aplikasi, tampilan animasi, manfaat aplikasi, dan materi yang digunakan dalam aplikasi. Pengujian ini menggunakan kuisioner untuk mendapatkan hasilnya. Kuisioner diisi oleh siswa SMP N 4 Tambang dan mualaf. Pada pengujian berikut, diberikan kuisioner kepada 15 siswa dan 2 orang mualaf. Untuk Tabel nama responden siswa dan mualaf dapat dilihat pada lampiran E.

Kuesioner yang telah dijawab oleh responden, nantinya akan dihitung persentasenya menggunakan rumus (1), sesuai yang di kemukakan oleh Ali (1998:134) yaitu :

$$P = (f/N) * 100 \% \dots\dots\dots (1)$$

dimana:

P = Jumlah persentase dimana dalam kuisioner ini, penulis mengkondisikan jawaban Sangat Setuju, Setuju, Kurang Setuju dan Tidak Setuju

f = frekuensi jawaban responden

N = jumlah tetap (dalam aspek ini adalah jumlah jawaban secara keseluruhan, untuk siswa adalah  $15 \times 5 = 75$  dan untuk mualaf adalah  $2 \times 5 = 10$ )

Setelah data dipersentase kemudian dianalisa dengan kriteria yang dikemukakan Ali (1998:184), yaitu :

|         |                           |
|---------|---------------------------|
| 100%    | : Seluruhnya              |
| 99%-76% | : Sebagian besar          |
| 75%-51% | : Lebih dari setengahnya  |
| 50%     | : Setengahnya             |
| 49%-26% | : Kurang dari setengahnya |
| 25%-1%  | : Sebagian kecil          |
| 0%      | : Tidak Ada               |



### 5.2.2.1 Hasil Kuisisioner Responden Siswa

**Tabel 5.4 Hasil Kuisisioner Responden Siswa**

| No           | Pertanyaan                                   | Sangat Setuju | Setuju    | Kurang Setuju | Tidak Setuju |
|--------------|--|---------------|-----------|---------------|--------------|
| 1            | Tampilan aplikasi menarik                    | 13            | 2         | 0             | 0            |
| 2            | Mudah menggunakan aplikasi Shalat            | 13            | 2         | 0             | 0            |
| 3            | Video 3D dan suara pada aplikasi sudah jelas | 13            | 2         | 0             | 0            |
| 4            | Aplikasi Memuaskan                           | 13            | 2         | 0             | 0            |
| 5            | Siswa suka menggunakan aplikasi              | 13            | 2         | 0             | 0            |
| <b>Total</b> |  | <b>65</b>     | <b>10</b> | <b>0</b>      | <b>0</b>     |

Dari hasil perhitungan pada table 5.3 diatas, dengan menggunakan rumus (1) sehingga untuk masing-masing kondisi diperoleh presentase sebagai berikut:

1.  $P_{\text{sangat setuju}} = (65/75) * 100\% = 86,67 \%$
2.  $P_{\text{setuju}} = (10/75) * 100\% = 13,33 \%$
3.  $P_{\text{kurang setuju}} = (0/75) * 100\% = 0 \%$
4.  $P_{\text{tidak setuju}} = (0/75) * 100\% = 0 \%$

Dilihat dari hasil persentase diatas dapat kita lihat bahwa persentase untuk kategori sangat setuju menempati posisi tertinggi dengan nilai 43,3% yang masuk kategori lebih dari setengahnya sangat setuju dengan aplikasi shalat.

### 5.2.2.2 Hasil Kuisisioner Responden Mualaf

**Tabel 5.5 Hasil Kuisisioner Responden Mualaf**

| No | Pertanyaan                                       | Sangat Setuju | Setuju | Kurang Setuju | Tidak Setuju |
|----|--|---------------|--------|---------------|--------------|
| 1  | Tampilan aplikasi menarik                        | 2             | 0      | 0             | 0            |
| 2  | Mudah menggunakan aplikasi Shalat                | 2             | 0      | 0             | 0            |
| 3  | Bisa mendengar dan melihat jelas video shalat 3D | 2             | 0      | 0             | 0            |

|              |  |           |          |          |          |
|--------------|--|-----------|----------|----------|----------|
| 4            | Dapat memahami dan mengerti materi aplikasi shalat | 2         | 0        | 0        | 0        |
| 5            | Aplikasi memuaskan                                 | 2         | 0        | 0        | 0        |
| <b>Total</b> |  | <b>10</b> | <b>0</b> | <b>0</b> | <b>0</b> |

Dari hasil perhitungan pada table 5.4 diatas, dengan menggunakan rumus (1) sehingga untuk masing-masing kondisi diperoleh presentase sebagai berikut:

1.  $P_{\text{sangat setuju}} = (10/10) * 100\% = 100\%$
2.  $P_{\text{setuju}} = (0/10) * 100\% = 0\%$
3.  $P_{\text{kurang setuju}} = (0/10) * 100\% = 0\%$
4.  $P_{\text{tidak setuju}} = (0/10) * 100\% = 0\%$

Dilihat dari hasil persentase diatas dapat kita lihat bahwa persentase untuk kategori setuju menempati posisi tertinggi dengan nilai 100% yang masuk kategori lebih dari setengahnya setuju dengan aplikasi shalat.

## **BAB VI**

### **PENUTUP**

#### **6.1 Kesimpulan**

Berdasarkan analisis, perancangan dan implementasi pada aplikasi yang telah dibuat dapat diambil kesimpulan sebagai berikut:

##### **6.1.1 Kesimpulan pengujian pada device android**

1. Aplikasi yang dibangun berhasil berjalan pada *smartphone* android dengan versi android diatas 2.2 (*froyo*).
2. Tampilan aplikasi menyesuaikan dengan besarnya layar pada *smartphone*.
3. Hasil implementasi sesuai dengan perancangan.

##### **6.1.2 Kesimpulan pengujian kuesioner**

1. Aplikasi yang dibangun sangat mudah digunakan (*user friendly*).
2. Tampilan video dan suara pada aplikasi sudah jelas.

#### **6.2 Saran**

Berikut ini adalah beberapa saran yang dapat digunakan sebagai saran untuk landasan pengembangan selanjutnya:

1. Dalam membangun aplikasi disarankan agar membuat animasi 3D menjadi lebih baik lagi.
2. Aplikasi dapat dikembangkan dengan menambah beberapa fitur seperti adanya tata cara wudhu sebelum shalat dalam bentuk video 3D dan doa-doa sesudah shalat.

## DAFTAR PUSTAKA

Abdullah Jibrin, Al-'Allamah, *Tuntunan Shalat menurut Al-Quran dan As-Sunah*, Yayasan Al-Sofwa

Adobe Photoshop, <http://id.wikipedia.org>

Al-Sya'rawi, Syaikh Mutawalli, *Tirulah Shalat Nabi!*, PT. Mizan Pustaka, 2008

*Android and iOS Surge to New Smartphone OS Record in Second Quarter*, <http://www.idc.com>, 8 Agustus 2012

Aripurnamayana, M. Irfan, *Rancang Bangun Pembuatan Mobile Learning Berbasis Android*, 2011

Darwiyanti, Sri dan Romi Satria Wahono. *Pengenalan Unified Modeling Language*. <http://ilmukoputer.org/2006/08/05/pengantar-uml/>, 14 Agustus 2012

Hardiansyah, Bangkit, *Informasi Keaneragaman Batik Pulau Jawa Berbasis Android*, 2012

Karolin, *Aplikasi Mobile Doa dan Dzikir Berbasis Android 2.2*, 2012

Nashiruddin Al-Albani, Syaikh Muhammad, *Sifat Shalat Nabi SAW Tata Cara Shalat Sesuai Dengan Tuntunan Rasulullah SAW*, AT-TIBYAN, Solo, 2012

Nashiruddin Al-Albani, Syaikh Muhammad, *Sifat Shalat Nabi Jilid 1*, Griya Ilmu, Jakarta, 2008

Nashiruddin Al-Albani, Syaikh Muhammad, *Sifat Shalat Nabi Jilid 2*, Griya Ilmu, Jakarta, 2008

*Penuntun Sholat Lengkap disertai Dzikir dan Wirid*, Mandala, Bandung, 2010

Rifa'I, Drs. Mohammad, *Risalah Tuntunan Shalat Lengkap*, PT. Karya Toha Putra, Semarang, 2010

Rational Team. *Rational Unified Process : Best Practices for Software Development Teams*, 2001.

Safaat H, Nazruddin, *Pemograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*, Informatika, Bandung, 2011

*Sistem Operasi Android*, <http://id.wikipedia.org>

*Unified Modeling Language*, <http://id.wikipedia.org>

Utamu, Puji, *Implementasi Mobile Learning Do'a Harian untuk Anak Pra Sekolah Berbasis Android*, 2012

Welkis. "3Dstudio MAX. dalam 3Dstudio MAX" [Online] Available  
[http://www.yaswarau.com/index.php?option=com\\_content&view=article  
&id=531:-3dstudio-max&catid=50:iptek&Itemid=201](http://www.yaswarau.com/index.php?option=com_content&view=article&id=531:-3dstudio-max&catid=50:iptek&Itemid=201), diakkses 05  
November 2011